EXHIBIT H-2

AMERICAN ARBITRATION ASSOCIATION AND

INTERNATIONAL CENTRE FOR DISPUTE RESOLUTION

COMMERCIAL ARBITRATION RULES

Bosch Automotive Service Solutions Inc.

      Claimant,

        Case Number: 01-21-0016-2306

   -vs-

        Arbitrator: Thomas W. Cranmer

Collision Sciences Inc.

      Respondent.

AUDIT REPORT

My name is Joshua HelfinSiegel. I am over the age of 18, have personal knowledge of the matters set forth in this Audit Report, and if called as a witness, I would be competent to testify to the matters in this Audit Report. The facts stated in this report are true and correct to the best of my knowledge and experience.

<div align="center">

Hrg. EX.

39

</div>

I. TABLE OF CONTENTS

II.   TABLE OF ATTACHMENTS

Attachment 1   CV of Joshua HelfinSiegel

Attachment 2   Materials Considered

Attachment 3   Release Dates of Bosch CDR Software

Attachment 4   Video Recordings of Audit

Attachment 5   List of Search Terms used during Audit

III. QUALIFICATIONS AND COMPENSATION

1.   I, Joshua HelfinSiegel, am above the age of 18 years old and the following statements are based on my professional knowledge and personal experience.

2.   I have a Bachelor's degree in Computer Science and a Certificate in Bioinformatics and Modeling from Wesleyan University, and have been an EnCase Certified Examiner in computer forensics since 2012.

3.   I am currently employed by DisputeSoft as a Systems Administrator and I.T. Consultant. I have over sixteen years of experience in IT systems administration, database management, security, software and hardware support. Additionally, I have served as a technical and forensic investigation consultant in various complex litigation matters for more than ten years at DisputeSoft, with a focus on matters involving intellectual property rights and software implementation failures.

4.   Particularly relevant here, I have performed forensic analyses, source code comparisons, and audits for various types of intellectual property disputes over many years working at DisputeSoft, searching for evidence of copying or misuse related to alleged trade secret misappropriation, copyright infringement, patent infringement, and breach of license claims. I have searched for evidence of literal and non-literal copying, as well as for the presence or absence of trade secrets and patented systems or methods within source code. My practical experience is described in my curriculum vitae, a true and correct copy of which is attached and incorporated hereto as Attachment 1.

5.   I am being compensated at the rate of $390 per hour for the work I have performed on behalf of Bosch Automotive Service Solutions ("Bosch") for this matter. My compensation is in no way contingent on my findings herein or on the outcome of this matter.

6.   The findings in this report are factual in nature. As such, I do not intend to express expert opinions in this report, though I may be called upon to testify as an expert witness in this matter at a later date. I have been assisted in this matter by DisputeSoft personnel. All analysis and other assistance in connection with the preparation of this report was performed and provided by me or by DisputeSoft staff under my supervision and direction. References to "I" or "me" refer to both myself and the staff members who assisted me in preparing this report.

IV.   SCOPE OF WORK

7.   I have been retained by Bosch Automotive Service Solutions Inc. ("Bosch" or "Claimant") to provide expert services in conducting an audit of Collision Sciences Services Inc.'s ("CSI" or "Respondent") use of Bosch's CDR Retrieval Tool Software ("Bosch CDR Software") (the "Software Audit"). This Software Audit is separate and apart from any software expert witness services I have been asked to provide in this arbitration. I understand that the findings I detail below may be used to support whether CSI has used the Bosch CDR Software in violation of the End User License Agreement ("EULA") applicable to CSI's license to the Bosch CDR Software and one or more of the other allegations against Respondent in this Arbitration.

8.   Specifically, I have been requested to perform the Software Audit by reviewing and analyzing CSI's corporate computer, software code repositories and cloud storage locations (collectively the "Audit Systems"). In my review of the Audit Systems, I have been requested to do the following:

8.1.   Search for evidence indicating the presence and/or use of the Bosch CDR Software by Respondent in the Audit Systems;

8.2.   Search for evidence in the Audit Systems of Respondent "reverse engineering" the Bosch CDR Software;

8.3.   Search for evidence in the Audit Systems to determine how Respondent's Crash Scan software is updated with new vehicles, insofar as whether that process utilizes data extracted from the Bosch CDR Software or the Bosch CDR Software itself;

8.4.   Index and search the Audit Systems for a shared list of keywords, then review the results as per relevance to the above three tasks; and

8.5.   Supplement my findings based on any additional evidence or reports provided on behalf of Respondent.

9.   Pursuant to Paragraph 10 of the April 20, 2022 Order on Software Audit, I captured all activities performed on the Audit Systems using freely-available Open Broadcaster Software for screen recording.[1] I periodically refer herein to the capture videos using the following designation: [hour:minutes] in the [MM-DD-YYYY] Capture Video. These videos are provided in native format as Attachment 4 to this report.

---

[1] More information on Open Broadcaster Software available at https://obsproject.com.

V.   MATERIALS CONSIDERED

10.   In conducting the Software Audit, I have reviewed and considered the following

materials:

10.1.   The Demand for Arbitration, dated August 20, 2021;

10.2.   Order on Software Audit, dated April 20, 2022;

10.3.   Respondent's Explanations Pursuant to Paragraph 8 of Audit Protocol Order;

10.4.   The End User License Agreements ("EULA") for versions 17.3 – 17.7 of the

Bosch CDR Software (BOSCH002827);

10.5.   The EULA for versions 17.8 – 18.1 of the Bosch CDR Software

(BOSCH002795);

10.6.   The End User License Agreements ("EULA") for the Bosch CDR Software,

dated May 30, 2019 (BOSCH000082-87);

10.7.   Produced documents:

10.7.1.   CS00437680-694 ("Development Notes");

10.7.2.   CS00016681-83 ("Text of CDR.au3"); and

10.7.3.   CS00444978 and CS00013052, emails referencing a "cdr-replay-

controller" repository;

10.8.   Crash Data Group Sales Receipts to Respondent for one year subscription to

Bosch CDR Software dated October 27, 2015 (BOSCH000923), January 26, 2018

(BOSCH000924) and July 9, 2019 (BOSCH000105);

10.9.   Remote access to the Audit Systems, comprised of:

10.9.1.   What I believe to be a clone of a laptop computer system used by

Brian Hsu, developer for Respondent (the "Laptop");

10.9.1.   Three source code repositories, hosted on Bitbucket for CSI, named

"cdrservice"; "cdr-bluetooth-app-ios"; and "cdrbluetoothapp-android";

collectively the "CSI Code Repositories";

10.9.2.   An Amazon Web Services account of Collisions Sciences, including

systems and databases hosted in Ohio, Northern California, and Canada;

10.10.   A complete listing of search terms used during the Audit are included as

Attachment 5;

10.11.   The video recordings of the Audit are included in native format as Attachment

4; and

11.   The documents listed in Attachment 2 attached hereto and footnote citations throughout

this report.

12.   My findings are based upon search, review, and analysis of these materials, as well as

my education, training, and experience, to conduct analyses and reach the findings detailed in

this report with a reasonable degree of professional certainty.

VI.   SUMMARY OF FINDINGS

13.   Based upon my review and analysis, as well as my education, training and experience, I have discovered, as further described in the remainder of this report:

13.1.   **The BOSCH CDR Software was installed on the Laptop.**

13.1.1.   The audit revealed that versions of the Bosch CDR Software were installed at least 325 times.[2] The installations included at least versions 16.4, 19.4, 19.4.2, 19.5, 19.6, 21.0, 21.1, 21.2, 21.3, 21.4, and 21.5 of the Bosch CDR Software. There was likely an automated process to install various versions of the Bosch CDR Software on the Laptop at regular intervals.

13.2.   **Respondent incorporated the Bosch CDR Software into their own "CDR Replay" tool.**

13.2.1.   The "CDR Replay" tool was used to mock-up the information sent to and from the Bosch CDR Software that occur during a car crash scan, without actually needing to connect to a real vehicle. CDR Replay requires the Bosch CDR Software as a critical piece to function.

13.2.2.   Respondent also used a bus-sniffing tool named "BUSMASTER" (coincidentally also created by a Bosch-affiliated entity) to intercept and view the messages sent to and from the Bosch CDR Software. This information could be used to attempt to reverse engineer and extract data transmitted by the Bosch CDR Software. I found a video demonstrating this process by the "CDR Replay" tool

---

[2] *See* install logs for the Bosch CDR Software discovered in C:\Windows\appcompat\Programs\Install; dtSearch hits from 345 to 670 are each install logs for the Bosch CDR Software; starting at approximately 4:24 PM in 8-16-2022 Capture Video.

on the Laptop.[3]

13.2.3. The C++ source code files related to BUSMASTER and CDR Replay were not reviewed during the Audit.[4]

13.2.4. Scripts used to automate the running of the Bosch CDR Software were found on the Laptop.

13.3. **There is evidence on the Laptop related to potential reverse engineering of the Bosch CDR Software.**

13.3.1. The CDR Replay tool including BUSMASTER are indicative of potential reverse engineering of the Bosch CDR Software. Additionally, memory dumps from the Python development environment, "PyCharm" also indicate potential reverse engineering of the Bosch CDR Software.

13.4. **Respondent Installed and Used Bosch CDR Software Licensed to Others.**

13.4.1. The Laptop contained Bosch CDR Software license files for several versions of the Bosch CDR Software licensed to companies that were not Respondent, namely Street Delivery and Biologic Forensics.[5]

13.4.2. The Bosch CDR Software version 21.5.1 licensed to "StreetDelivery" was run on the Laptop at least on one occasion, on July 19, 2022.[6]

13.5. **Respondent's source code repositories made available to me did not include**

---

[3] This video was located in C:\Users\Brian\Documents\Expression\Expression Encoder\Output\Brian-MBP 4-3-2018 2.20.25 PM\CDR Replay.wmv and can be viewed at approximately 3:35 pm in the 8-24-2022 Capture Video

[4] I attempted to review the contents of these files on May 15, 2023 at approximately 2:43 pm, but it appeared that the Laptop may have been in use, so I disconnected from the Laptop.. I was able to validate the presence of these files by reviewing the videos recorded during the Audit, but I was not aware of their potential relevance at that time.

[5] Located in the directories C:\Users\Brian\Documents\CDR Installation\ and C:\Users\Brian\Documents\CDR Installation\Supplier\; reviewed in videos 8-16-22 at approximately 3:21 pm; 8-25-2022 Capture Video at approximately 1:16 pm and 3:47 pm.

[6] *See* crash dump file C:\Users\Brian\AppData\Local\CrashDumps\CDR.EXE.9656 on July 19, 2022; reviewed in 8-25-2022 Capture Video at approximately 12:54 pm.

a repository for a "cdr-replay-controller."

13.6.   "CS_Production" database appears to be based upon and expanded from the one shown in the April 3, 2018 "CDR Replay.wmv" demonstration video for "CDR Replay."

VII.   FACTS AND BACKGROUND INFORMATION

A. Bosch's CDR Software

14.   Bosch's marketing material states that since the year 2000, Bosch has used its CDR

Software to retrieve data from Event Data Recorders ("EDR") installed as part of airbag or

other safety systems on automotive vehicles. The Bosch CDR Software has been used to read

the data stored on the EDR inside the vehicle, and provide CDR reports based on the data

extracted from a vehicle after a car crash.[7]

15.   By default, the Bosch CDR Software installs in a read-only mode; to unlock all of the

function of the Bosch CDR Software, including vehicle scans, printing and saving Bosch-

branded CDR reports, a user must purchase a subscription for a one-year license.[8] More

information on the software installation process, certificates, subscriptions and activation is

available in the "Software Installation Guide" for the Bosch CDR Software.[9]

B. Respondent's Crash Scan Software

16.   CrashScan by CSI is marketed as a "universal, mobile-app-based EDR (Event Data

Recorder) solution" that also includes "software required to extract all digital forensic data

from supported vehicles," user training, and EDR Reports.[10]

17.   A review of the code provided in CSI's online Bitbucket source code repositories

revealed that the repositories appear to contain code related to Respondent's Crash Scan

application, broken up into three repositories: the Server Application, which could be

considered the engine of the application, appears to perform the majority of the functions of the

---

[7] *See* https://cdr.boschdiagnostics.com/cdr/sites/cdr/files/15-93_cdr_crash_data_retrieval.pdf.
[8] Demand for Arbitration, p. 7.
[9] BOSCH002655.
[10] *See* https://www.collision-sciences.com/services.html.

application and contains the API; the Android application, which appears to use the API to access the functionality of the Server Application; and the iOS Application, which also appears to use the API to access the functionality of the Server Application.[11]

C. Relevant Licenses

18.   Based on receipts I have seen from Crash Data Group it appears that Respondent purchased one-year licenses for the Bosch CDR Software on October 27, 2015, January 26, 2018, and July 9, 2019.[12] A one-year license allows the user to access and use the then current version of the software, and all future versions that are released during the one-year license term.[13]

19.   I have reviewed the website at https://cdr.boschdiagnostics.com/cdr/software-downloads, where information on official versions of the Bosch CDR Software are available, and more recent versions are available for download. The release date for each version is provided when a user clicks on a given software version number on the website, e.g., 19.6. The release dates gathered from the Bosch CDR Software download pages for each version of the Bosch CDR Software released since the date of Respondent's first license purchase are reflected in the table below, as well as in Attachment 3.

Table 1: Bosch CDR Software Versions and Release Dates show on Bosch website

Bosch CDR Software Version	Date of Release	Bosch CDR Software Version	Date of Release
v23.1	03/31/2023	v21.2	8/3/2021
v23.0	10/7/2022	v21.1	5/14/2021
v21.5	4/8/2022	v21.0	1/21/2021
v21.4	1/5/2022	19.6	12/1/2020
v21.3	10/28/2021	v19.5	8/13/2020

---

[11] *See* also Item 6(c) of Respondent's Explanations Pursuant to Paragraph 8 of Audit Protocol Order.
[12] *See* BOSCH000923, BOSCH000924, and BOSCH000105, respectively.
[13] *See* https://cdr.boschdiagnostics.com/cdr/products/cdr-system-1-year-software-subscription.

Bosch CDR Software Version	Date of Release	Bosch CDR Software Version	Date of Release
v19.4	5/7/2020	v17.5	10/17/2017
v19.3	12/23/2019	v17.4	6/15/2017
v19.2	12/6/2019	v17.3	4/11/2017
v19.1	9/20/2019	v17.2	1/22/2017
v19.0	6/6/2019	v17.1	11/2/2016
18.0	2/20/2019[14]	v17.0	8/23/2016
v17.10	12/20/2018	v16.6	5/23/2016
v17.9	9/18/2018	v16.5	3/18/2016
v17.8	7/31/2018	v16.4	12/17/2015
v17.7	3/19/2018	v16.3	11/20/2015
v17.6	12/22/2017	v16.2	8/10/2015

20.   Given the release dates above and that Respondents purchased one-year licenses from October 27, 2015 to October 27, 2016; January 26, 2018 to January 26, 2019; and July 9, 2019 to July 9, 2020, the versions available during those time periods were versions 16.2 – 17.0; 17.6 – 17.10; and 19.0 – 19.4 of the Bosch CDR Software.

21.   Based on the above and discussions with counsel, it is my understanding that three EULAs are relevant to Respondent's use of the Bosch CDR Software: 1) the EULA for versions 17.3 – 17.7; 2) the EULA for versions 17.8 – 18.1; and 3) the EULA dated May 30, 2019 for versions 19.0 and up.[15]

---

[14] There appears to be a typo in the release year for version 18.0, as the previous version, 17.10, was release on December 20, 2018, and the software release details document shows a copyright notice in 2019.
[15] *See* BOSCH002827-831, BOSCH002795-799, and BOSCH000082-087.

VIII.   FINDINGS AND ANALYSIS

A.   The Bosch CDR Software is installed and was run on the Laptop

22.   The Bosch CDR Software is present and currently installed on the Laptop at the location: C:\Program Files (x86)\Bosch\Crash Data Retrieval. As of August 14, 2022, the currently installed version on the Laptop was 19.4. Log files on the Laptop show that version 19.4 of the Bosch CDR Software was installed on May 27, 2022.[16]

23.   Additionally, evidence on the Laptop indicates that multiple versions of the Bosch CDR Software were installed and/or reinstalled at least 325 times.[17] Setup logs in the Windows "temp" directory indicate installations as recent as August 14, 2022, after the audit had started, as in the log file C:\Users\Brian\AppData\Local\Temp\Setup Log 2022-08-14 #001.txt. There are installation logs present in the "temp" directory on the Laptop for the following eleven versions of Bosch's CDR Software on August 14, 2022:

Table 2 Versions of Bosch CDR Software that had been installed on the Laptop as of 8-14-2022

16.4	19.4	19.4.2	19.5	19.6	21.0
21.1	21.2	21.3	21.4	21.5	

24.   The Laptop also contains evidence that the Bosch CDR Software was run. When the Windows operating system or a Windows program crashes, it often will save records of what the state of the computer was and what was loaded into the computer's memory when the crash occurred. This information is often saved into crash "dump"[18] or DMP files, and the

---

[16] *See* INSTALL_ffff_b161e7f0-1bd2-47e6-b841-0df61bdf47ef.txt located here:
C:\Windows\appcompat\Programs\Install\; as seen at approximately 2:58 pm in the 8-16-22 video.

[17] Install logs for Bosch CDR Software discovered in C:\Windows\appcompat\Programs\Install; dtSearch hits from 345 to 670 are each install logs for the Bosch CDR Software; approximately 4:24 PM in 8-16-2022 Capture Video.

[18] A memory dump can be defined as, "a display of the contents of all or part of a computer's internal storage, usually in binary, octal, or hexadecimal form," per (ISO/IEC/IEEE 24765:2017 Systems and software engineering-Vocabulary) as reviewed at https://pascal.computer.org/sev_display/search.action;jsessionid=0fhfn-A0emlVW0a280XPoPNl-HQK4jC9BJvJjudV.cslcpav04.

information is useful for possible future debugging of why Windows or a Windows application crashed.[19]

25.   Crash dump files on the Laptop indicate that the Bosch CDR Software was open and running when the Windows Operating System crashed on January 11, 2022.[20] Four additional crash dump files indicate different dates between May 21, 2022, and August 11, 2022 when application crashes occurred with the Bosch CDR Software open and running on the Laptop.[21]

26.   I found additional evidence of crashes and hangs of the Bosch CDR Software on the Laptop in Windows error report files located in the directory C:\Users\All Users\Microsoft\Windows\WER\ReportArchive.[22] An application crash is when a program encounters an error and needs to close unexpectedly.[23] An application hang, as differentiated from a crash, is when a Windows application becomes unresponsive for a period of time, but does not crash, and eventually becomes responsive again.[24] This directory on the Laptop contains evidence of four application hangs and 82 application crashes for the Bosch CDR

---

[19] An application crash can be defined as, "a sudden and complete failure of a computer system or component," per (ISO/IEC/IEEE 24765:2017 Systems and software engineering-Vocabulary), reviewed at https://pascal.computer.org/sev_display/search.action;jsessionid=0fhfn-A0emlVW0a280XPoPNl-HQK4jC9BJvJjudV.cslcpav04.

[20] *See* C:\Windows\MEMORY.DMP, as shown at approximately 2:51 pm in 8-16-22 video.

[21] *See* (1) C:\Windows\LiveKernelReports\DripsWatchdog-20220521-1319.dmp, on May 21, 2022 as shown at approximately 2:07 pm in the 8-16-2022 Capture Video; (2) C:\Users\Brian\AppData\Local\CrashDumps\CDR.EXE.9656, on July 19, 2022 as shown at approximately 2:42 pm in the 8-16-22 video; (3) C:\Users\Brian\AppData\Local\CrashDumps\CDR.EXE.13276 on July 26, 2022 as shown at approximately 2:48 pm in the 8-16-2022 Capture Video; and (4) C:\Users\Brian\AppData\Local\CrashDumps\CDR.EXE.13464 on August 11, 2022 as shown at approximately 2:51 pm in the 8-16-2022 Capture Video.

[22] More information on Windows Error Reporting ("WER") available at: https://learn.microsoft.com/en-us/windows/win32/wer/windows-error-reporting.

[23] "A crash is when something experiences a fault and has no choice but to exit," as described at https://techcommunity.microsoft.com/t5/ask-the-performance-team/basic-debugging-of-an-application-crash/ba-p/372392.

[24] "The operating system defines an application hang as a UI thread that has not processed messages for at least 5 seconds. Obvious bugs cause some hangs, for example, a thread waiting for an event that is never signaled, and two threads each holding a lock and trying to acquire the others. You can fix those bugs without too much effort. However, many hangs are not so clear. Yes, the UI thread is not retrieving messages - but it is equally busy doing other 'important' work and will eventually come back to processing messages," available at https://learn.microsoft.com/en-us/windows/win32/win7appqual/preventing-hangs-in-windows-applications.

Software ("CDR.EXE"), spanning from July 7, 2021, through August 11, 2022 as described in

Windows error report files (Report.wer files). These hangs and crashes are predominantly for

version 17.9 and 19.4 of the Bosch CDR Software with at least one crash of version 21.5.1. [25]

27.　　Additional related evidence to Bosch CDR Software usage includes the Internet

browser history from Google Chrome, indicating visits to the website related to Bosch CDR

Software downloads for versions 17.5, 17.6, 17.10, and 18.0. The Internet history was located

at "C:\Users\Brian\AppData\Local\Google\Chrome\User Data\Default\History." and indicates

visits to https://cdr.boschdiagnostics.com/cdr/ websites related to Bosch CDR Software

downloads.[26]

28.　　Lastly, evidence found on the Laptop indicates that the default app for opening

"CDRx" files, electronic crash data report files, was "CDR.exe," the Bosch CDR Software.

This evidence was found in the Windows event log for "app defaults" on the Laptop, located in

C:\Windows\System32\winevt\Logs\Microsoft-Windows-Shell-Core%4AppDefaults.evtx.

B. Respondent has incorporated the Bosch CDR Software into their own "CDR Replay" tool.

29.　　Respondent has incorporated the Bosch CDR Software as part of their own "CDR

Replay" tool. This is apparent from an April 3, 2018 video demonstration of Respondent's

"CDR Replay" tool which I discovered on the Laptop.[27]

---

[25] As seen from approximately 3:08 pm to 3:42 pm in 8-16-2022 Capture Video.

[26] I understand from the August 20, 2021 Demand for Arbitration, p. 6, that versions of the Bosch CDR Software were made available on the website "https://cdr.boschdiagnostics.com/cdr/software-downloads", the same web address shown as visited at approximately 1:40 pm in the 8-19-2022 Capture Video. Also *see* website history URLs shown in 8-17-2022 Capture Video at 11:52 am, 12:00 pm, and 12:04 pm. For example, https://cdr.boschdiagnostics.com/cdr/cdr-v215-system-software; *see* 12:04 pm in 8-19-2022 Capture Video. Webpage as it existed can be viewed from the Internet Archive, available at: https://web.archive.org/web/20220520180901/https://cdr.boschdiagnostics.com/cdr/cdr-v215-system-software.

[27] This CDR Replay tool is also referenced in BOSCH002821, "Convert App-collected Cloud Crash Data —> Bosch CDR PDF (via Car simulator in Lab) *for any scans using the CrashScan mobile app, upon request, we can replay the data into the Bosch or Hyundai tool to generate the Bosch CDR PDF."

30.   I discovered what appears to be a demonstration video made on April 3, 2018, titled

"CDR Replay.wmv."[28] Based on the video, Bosch CDR Software appears to be necessary for

CSI's "CDR replay" tool to function. At a high level, the video demonstrates CSI's "CDR

Replay" tool in use, including incorporation of the Bosch CDR Software as part of the process

followed by Collision Sciences for its "CDR Replay." This video also demonstrates one

potential method for reverse engineering EDR data using the Bosch CDR application and CSI

tools, namely an "eavesdropping" software tool named "BUSMASTER" appears to listen for

and intercept messages sent to and from the Bosch CDR Software tool.

31.   In particular, CDR data is extracted from a database, formatted, and used as input to a

python program that appears to "replay" CDR crash data. During this "replay," a Bosch CDR

report is generated using the Bosch CDR Software "CDR.exe" version 17.7, while

BUSMASTER is "eavesdropping" on the messages sent to and from the Bosch CDR

Software.[29]

32.   Based on information within the video, the demonstration appears to have been

conducted on an Apple computer running Windows, as the computer name is listed as

"Bootcamp," a tool that can be used to run Windows on a Macintosh computer with an Intel

processor, and as the E: drive on the computer is listed as "Macintosh HD," a default name

often automatically given to the primary hard drive on an Apple computer.[30] Additionally, Item

6(a) of Respondent's Explanations Pursuant to Paragraph 8 of Audit Protocol Order lists the

second item as a MacBook Pro 13, dated "September 2017 to May 2021. Disposed as

---

[28] This video was located in "C:\Users\Brian\Documents\Expression\Expression Encoder\Output\Brian-MBP 4-3-2018 2.20.25 PM\CDR Replay.wmv"

[29] As seen at approximately 3pm in the 8-19-2022 Capture Video.

[30] *See* https://support.apple.com/en-us/HT208496, "Your built-in startup disk should be the first item listed in the Disk Utility sidebar. It's named Macintosh HD, unless you changed its name."

electronic waste in June 2021." This MacBook computer was likely used to create the "CDR Replay" video and run the "CDR Replay" tool, however, this computer was "disposed" in June of 2021, and it was not made available for examination.

33.   Further, information on the key components of source code related to the "CDR Replay" tool appear to be described in **C:\Users\Brian\Desktop\Desktop\SOURCE CODE\Summary of Code.pdf.**[31] There are three key components described within the PDF file, which appear to be located within the "C:\Users\Brian\Desktop\Desktop\SOURCE CODE\CDR and code related\" directory within folders named "cdr-service", "cdr-scripts", and "cdr-replay1", respectively.[32]

> 33.1.   "CDR-Replay": The purpose of the contents of this folder is to "listen for Parameter Identification ("PID") requests coming from a Bosch unit." Based on the description, this code would likely have been for handling the "eavesdropping" portion described above in conjunction with the BUSMASTER tool, and playing back "traces" recorded from an actual vehicle scan. If a source code repository for this source code exists on Bitbucket for the "CDR-Replay" tool, I was unable to locate it, or it has not been made available to me. A screenshot of the text related to "CDR-Replay" is included below.

---

[31] The "Summary of Code.pdf" file was last modified September 1, 2017 and was created on the Laptop on May 31, 2021. Details and contents of the file shown at approximately 2:45 pm in the 8-18-22 Capture Video.
[32] As shown at approximately 2:48 pm in the 8-18-22 Capture Video.

Image 1: Screenshot with search term highlighted of "CDR-Replay" portion of the "Summary of Code.pdf" file on Laptop

CDR Replay

The purpose of the contents of this folder is to listen for PID requests coming from a Bosch unit and through a sort and send back pre-recorded responses coming from data taken previously from a real car

The important contents here are one main script (main.go) and a few example JSON files.

The JSON files seem to be examples of how the data used to be stored (Stored as a file with the VIN, the Request and Response IDs, and the request PID and response value pairs) as opposed to the new way they're stored (in a database where the VIN is a column, the request response pair are a column and the request response pairs aren't stored but determined using the VIN)

To use the main.go script the Bosch unit must be hooked up to the computer via usb and again connected to the computer via a Cantact OBD2 to usb connector.

Once setup the main.go script is run with two inputs: the path of a JSON file in the old format (the same as the ones in the same folder), and the name of the port connected to a Cantact unit. The script loads the file and waits for incoming requests on the provided port and then finds and sends back the corresponding response from the input file. It will keep looping until manually shut down by the user.

33.2.   "CDR-Scripts": The purpose of this folder is to convert "traces" from outgoing requests from Bosch units into process JSON files containing necessary information for generating PDF reports and for automated playback in the "CDR-Replay" tool. If a source code repository for this "CDR-Scripts" source code exists on Bitbucket, I was unable to locate it, or it has not been made available to me.

Image 2: Screenshot with search term highlighted of "CDR-Scripts" portion of the "Summary of Code.pdf" file on Laptop

33.3.   "CDR-Service": The purpose of these files is server setup; run.py runs a server on the machine; config.py updated to match SQL database creds. The "Processes" folder contains JSON processes; the "app" folder contains API, api.py, models.py, and vehicle.py. Based on my review, this code is likely the precursor code to what eventually became the current "cdrservice" source code repository stored on Bitbucket.

Image 3: Screenshot with search term highlighted of "CDR-Scripts" portion of the "Summary of Code.pdf" file on Laptop

34.   The document describes, in essence, a three-part system: The first portion, the "CDR-Scripts," converts PIDs "traces" taken from vehicles into JSON data which can be used as input for "play back" in the "CDR Replay" tool to generate Bosch PDF crash reports and "eavesdrop" on information sent to and from the Bosch CDR Software and hardware during a replay of a crash scan. The second portion, the "CDR-Replay," plays back crash data from the converted "traces" into the Bosch CDR Software, listens to messages sent to and from Bosch CDR Software and hardware, and can generate Bosch CDR Software reports. The last portion, the "CDR-Service," is the engine for the online crash scan and data processing for CSI, which uses JSON data inputs to generate and email reports and other data.

35.   The process flow of the "CDR Replay" described here, appears to be to: 1) Acquire "trace" PID data from a real vehicle; 2) convert that "trace" data to JSON to prepare it for "play back" into the Bosch CDR Software using the "CDR-Scripts"; and 3) "Play back" that data, and eavesdrop on the messages sent to and from the Bosch CDR Software and hardware using the "CDR Replay" tool. Reports can be generated from the Bosch CDR Software in this way. It also appears that the converted JSON traces can serve as input to Respondent's "CDR-Service" for analysis and report generation as well. Examples of these reports appear to be present within a directory on the Laptop "C:\Users\Brian\Documents\Crash Data Retrieval\Generated via CDR Replay\"[33]

36.   Further, it appears that the "CDR Replay" process described above is demonstrated in a video found on the Laptop in an April 3, 2018 video titled "CDR Replay.wmv."[34] This video was extremely useful in viewing and understanding how Respondent's "CDR Replay" tool works. A detailed review of the April 3, 2018 "CDR Replay.wmv" video and description of what occurs is included below:

36.1.   The user first logs into a PostgreSQL database hosted at "app.collisionsciences.ca/phppgadmin/", then navigates to the "cdr" database. The "cdr" database in the video has only one table in it, labeled "cdr_data" with an estimated 430 rows. The "cdr_data" table contains numerous visible columns including: ID, date_created, date_modified, vin, vehicle,[35] cdr_data,[36] report_created (mostly FALSE), and email

---

[33] As seen at approximately 1:09 pm in the 8-19-2022 Capture Video.

[34] This video was located at "C:\Users\Brian\Documents\Expression\Expression Encoder\Output\Brian-MBP 4-3-2018 2.20.25 PM\CDR Replay.wmv". The "CDR Replay.wmf" file can be seen starting at approximately 3:35 pm in the 8-24-2022 Capture Video2022 Capture Video.

[35] This column appears to be the manufacturer, sometimes also year and make, e.g., Honda, 2012 Toyota PRIUS, 2014 Mazda Mazda3, etc.

[36] This column appears to be JSON process data; includes "process_type": "P_GETCRASHDATA", request id, protocol, response_id, DATE, etc.

(values including: jbayley@collisio[nsciences.ca], changyun.hsu@[gmail.com], devendrap@outloo[k.com], "NULL", and blank cells).

36.2.   The user then sorts the data by column ID number, and navigates to the latest entry on "Last" page. As of April 3, 2018, there are 1,057 unique IDs and rows generated for the "cdr_data" table. The user then selects the "cdr_data" field for a 2012 Toyota Camry, a row with unique ID 1056, the next-to-last row in the video.

36.3.   For this record, the email listed is changyun.hsu@gmail.com, and the "code" field (not shown on previous record) value is "Y." The visible JSON data starts with: {"max_front_lat_accel":[], "max_side_lat_acce." A screenshot from the April 3, 2018 video depicting this JSON data is included below.

Image 4: Visible Columns from Row ID 1056 in "cdr_data" from April 3, 2018 video
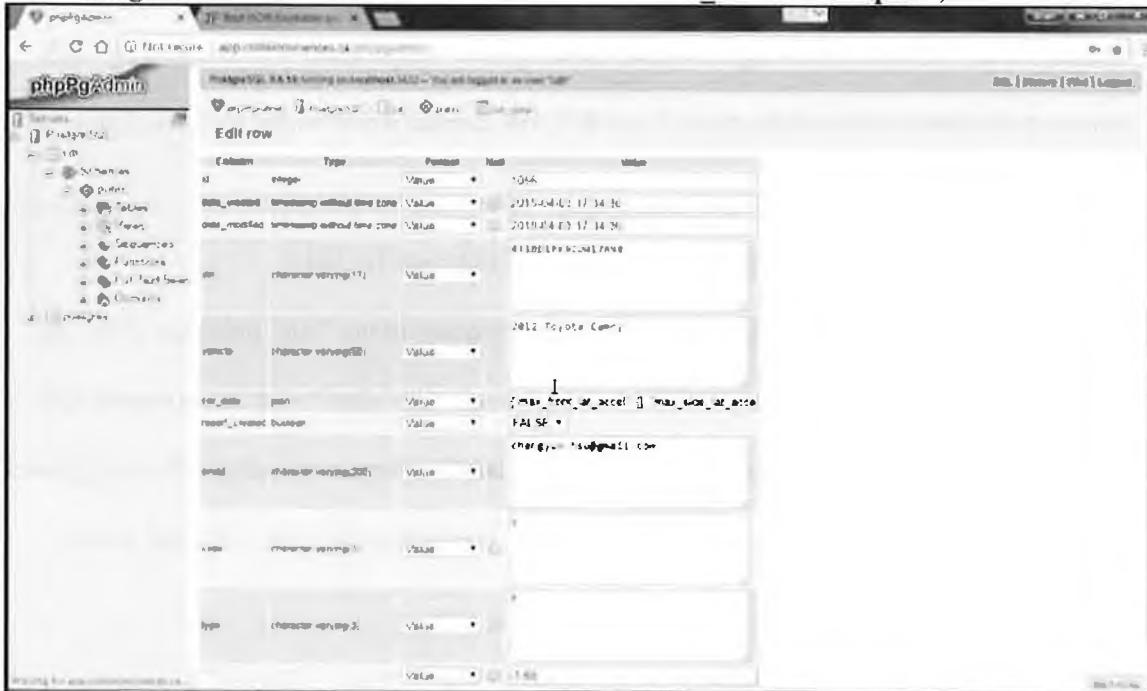
36.4.   The user then copies the JSON data from the "cdr_data" field in the database, and then pastes it into a JSON formatter hosted at https://jsonformatter.org. The user then formats the JSON data; the data is more clearly shown after the formatting. Fields

formatted shown include: max_front_lat_accel; max_side_lat_accel, response_id, max_roll_angle, request_id, max_roll_lat_accel, max_front_long_accel, max_roll_rate, part_type, process_type "P_GETCRASHDATA", have_crash_data, edr, and items.

36.5.   The video then appears to display files related to the "CDR Replay" tool, located in "C:\Users\Brian\PycharmProjects\JSON Processor":

36.5.1.   CDR Replay (folder shortcut);

36.5.2.   EEPROM.json;

36.5.3.   EEPROM.py;

36.5.4.   EEPROM.tx;

36.5.5.   Input.json;

36.5.6.   Processor.py;

36.5.7.   ReplayFileMaker.py;

36.5.8.   SWCAN Replay (folder shortcut).

37.   The computer name shown in the video is listed as "BOOTCAMP," which likely means the underlying system recorded to create the "CDR Replay.wmv" video was probably an Intel-based Macintosh computer running Microsoft Windows by using the "Bootcamp" software. Further support for this is that the secondary hard drive labelled "E" is named "Macintosh HD," a default hard drive name often used for the primary disk drive in Apple computers.[37]

38.   After the JSON data is formatted on the web page, the formatted JSON data is pasted into a local file on the computer: "C:\Users\Brian\PycharmProjects\JSON Processor\Input.json." The user then opens the code file:

---

[37] More information on Apple's "Bootcamp Assistant" available at https://support.apple.com/en-us/HT201468.

"C:\Users\Brian\PycharmProjects\JSON Processor\ReplayFileMaker.py"[38]

39.   Within "ReplayFilemaker.py" as shown in the video, the variable to hold data output is listed as "outputFID," and the output data is saved into "Replay.txt" per lines 10-11. The ReplayFileMaker.py file also references "request_id" and "response_id," setting them equal to the replay request_id and response ID.

40.   The user then runs "ReplayFileMaker.py," which saves its output into a new "Replay.txt" file. The user then copies and moves "Replay.txt" into the shortcut for the "CDR Replay" folder. The user overwrites an existing copy of "Replay.txt" within the "CDR Replay" folder. The user then opens "CDR Replay" shortcut from JSON Processor folder, which opens up to the "C:\users\Brian\My Documents\BUSMASTER\Code\CDR Replay" folder.

41.   The "CDR Replay" folder contains multiple files named after makes and models of vehicles. As the user renames the "Replay.txt" file, it is likely that the files here are renamed versions of "Replay.txt" files. Additional files are shown in the folder: CDR_Replay.cpp; CDR_REPLAY.def; CDR_Replay.dll; CDR_Replay.o; and CDR_Replaymake files. Based on names and extensions, these files appear to have been code files/libraries/executables written in C++ related to Respondent's "CDR Replay" tool.[39] The April 3, 2018 video also shows a "ReplayFileMaker" Microsoft Excel file, SWCAN.txt, and VIN.txt file within the "CDR

---

[38] Code shown within the "ReplayFileMaker.py" in the April 3, 2018 video appears similar to code within later "replay.py" code available within Respondent's source code repository located at "Cdrservice/elasticbeanstalk/app/replay.py" Line 18 in "replay.py" matches exactly the comment text "# File for data output" compared to line 10 from ReplayFileMaker.py from April 3, 2018 video (shown in August 24, 2022 Capture Video2022 Capture Video at 3:35 pm). Similarly, code on line 16 and 17 where "if request_id == 0x6f1:" and "# BMW. Extended Addressing mode," compared to lines 50-51 from replay.py. These suggest that replay.py is an update to ReplayFileMaker.py. The newer "replay.py" is longer hard-coded to write to "Replay.txt" but rather to "StringIO()" when comparing line 11 in "ReplayFileMaker.py" to line 19 in "replay.py". The end of the "replay.py" file also still shows "Replay.txt" and "Replay.json" as output options on lines 513-516; as shown in December 21, 2022 Capture Video2022 Capture Video at 1:41 pm.

[39] As shown at approximately 4:51 pm in the 8-26-2022 Capture Video; on the Laptop the aforementioned C++ code and related files all are last modified in August of 2022.

Replay" folder.

42.   The user opens "VIN.txt" and copies the VIN number from the 2012 Toyota Camry from the "cdr_data" table in the postgres database into the "VIN.txt" file as the first line.

43.   The user then opens the BUSMASTER application with the "CAN" tab selected, with config file loaded named "C:\Users\Brian\Documents\Bosch CDR.cfx." Notably, the Bosch CDR.cfx config file for the configuration of BUSMASTER used in the video is no longer present on the Laptop.[40] The bottom of the BUSMASTER screen displays the messages "CAN Recording," "J1939 Recording," and "PEAK USB 500.000 Kbps." The user then clicks the "Connect" button on BUSMASTER.

44.   The user then opens the Bosch CDR Software (version 17.7) and clicks on what appears to be a "New" icon. The user selects the "Brand" Toyota within the Bosch CDR Software, then clicks the "Read VIN from Vehicle" button. The Bosch CDR Software then displays "Attempting to read VIN from vehicle," and the BUSMASTER application displays the window named "Message Window – CAN" and begins to populate with Hexadecimal data. The VIN number is successfully retrieved and shown in the window named "Trace Window" in BUSMASTER, as well as in the Bosch CDR Software. Screenshots of this process taken from the "CDR Replay.wmv" movie file are included below, with BUSMASTER shown on the left, the Bosch CDR Software on the right.

---

[40] File is not present as shown by inspection of C:\Users\Brian\Documents\ in the 8-19-2022 Capture Video2022 Capture Video, at approximately 1:28 PM.

Image 5: CDR Replay Tool Reading VIN Number using BUSMASTER and Bosch CDR Software
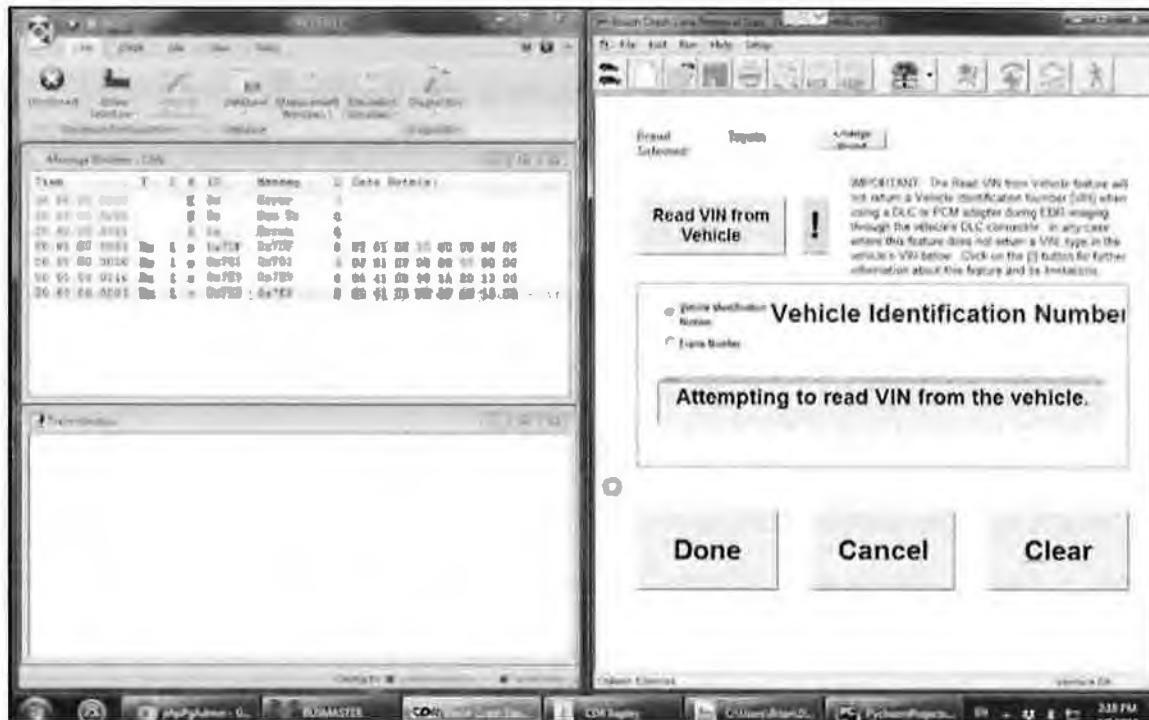
Image 6: CDR Replay Tool VIN Number Read using BUSMASTER and Bosch CDR Software
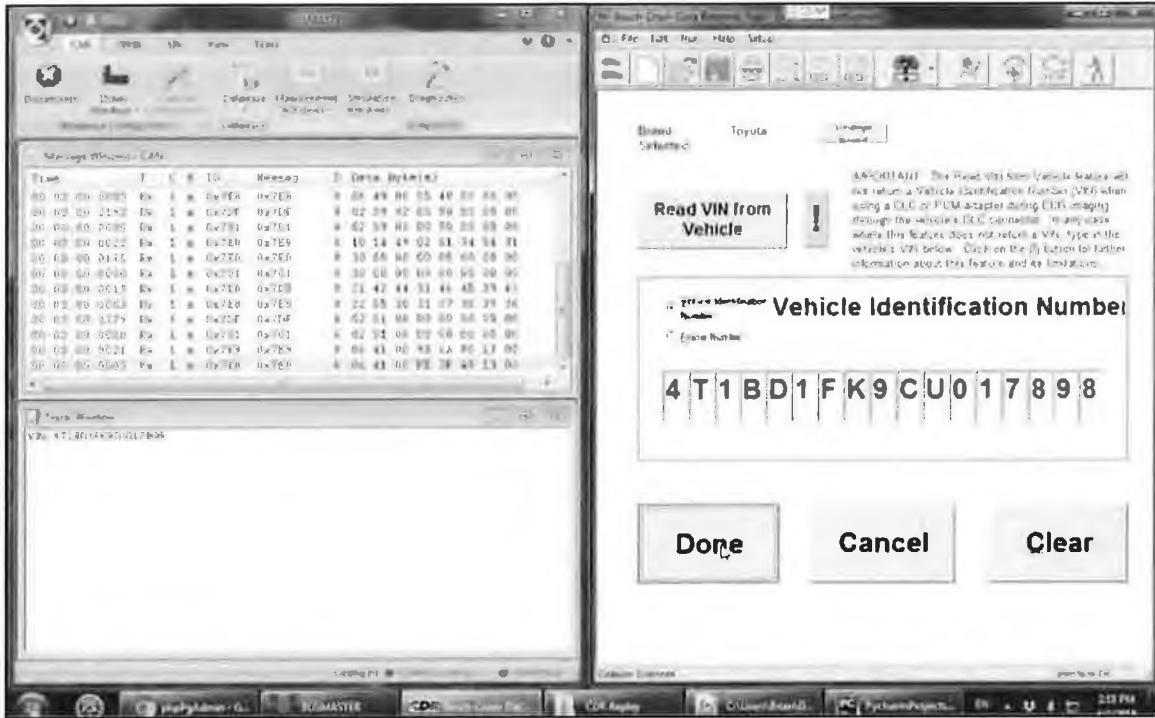
45.   The user clicks on the "Done" button on the Bosch CDR Software three times. The user then clicks on what appears to be a blue "Airbag" icon within the Bosch CDR Software. The Bosch CDR Software displays "Reading Data From Module," and "Pass 1." At the bottom of the window, text has changed to display "Collecting Data pass: 1." In the BUSMASTER "Trace Window," two lines have updated to display "Security Access" and two hexadecimal keys. The Bosch CDR Software performs three data collection "passes," and the data continues to update in the BUSMASTER "Message Window – CAN." When the data collection finishes, the Bosch CDR Software prompts the user if they want to "Save recovered data," and updates to the bottom of the window stating "Data Recovered," "Events: Front/Rear (2), Side (2)," and "Interface OK" as shown in the screenshots below.

Image 7: CDR Replay Tool Airbag Button Pressed (Pass 1)
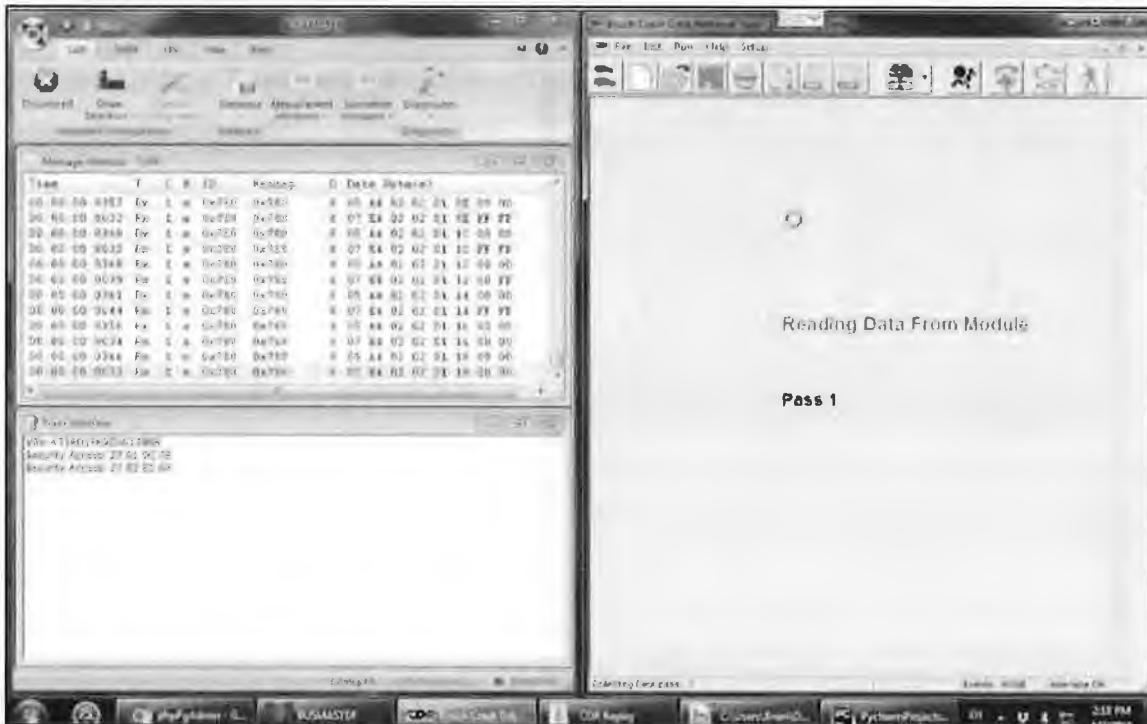
Image 8: CDR Replay Tool Airbag Button Pressed (Pass 3)
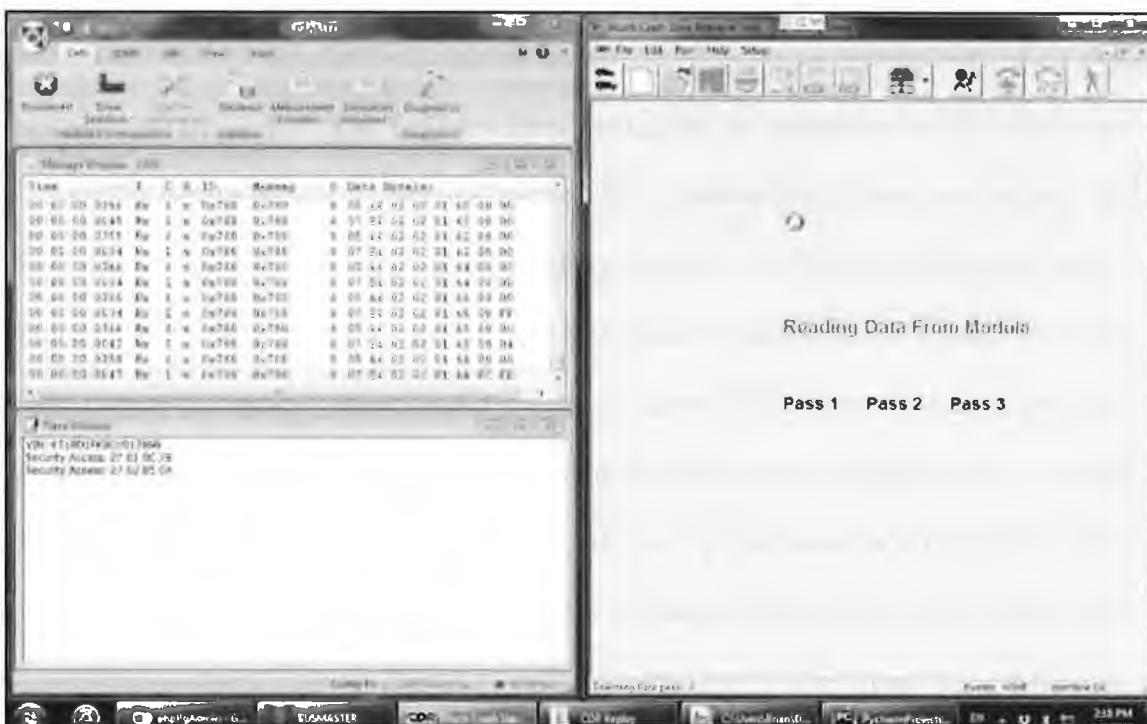
Image 9: CDR Replay Tool Airbag Button Pressed (Complete)
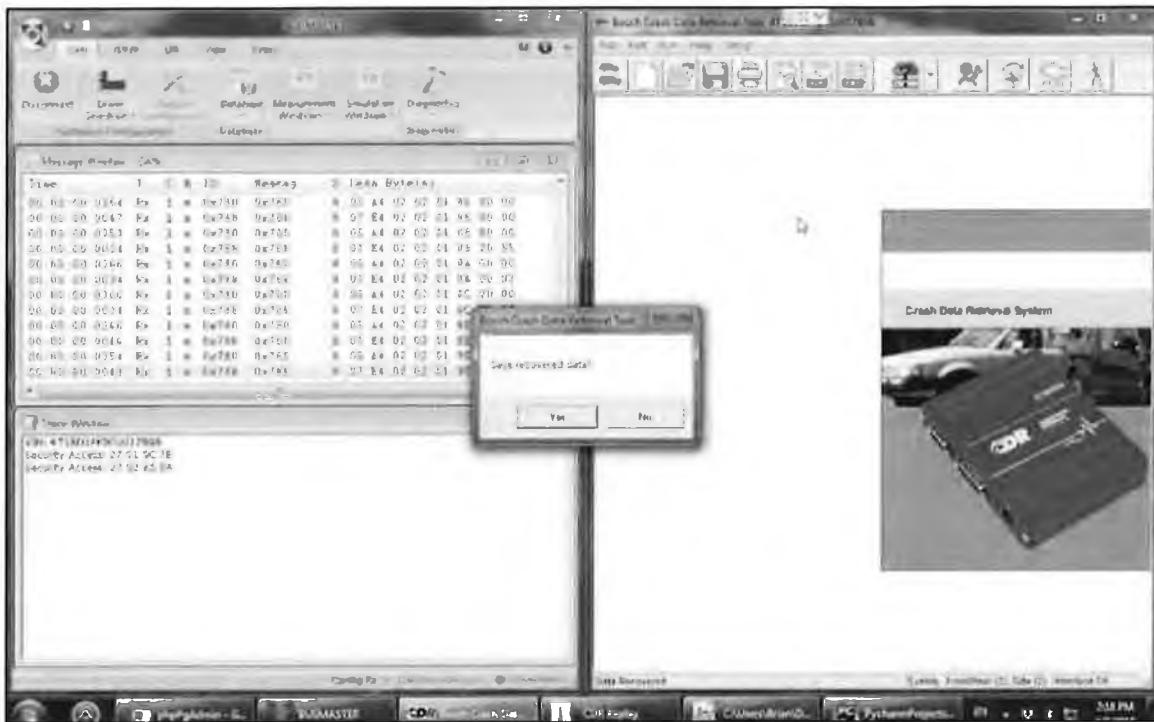
46.   The user then clicks "No" on the save data prompt. The Bosch CDR Software then generates a report Crash Data Retrieval Tool 17.7 licensed to Collision Sciences for the EDR Device "Airbag Control Module" with events recovered "Front/Rear (2), Side (2)" for the VIN number from the 2012 Toyota Camry listed in the "cdr" PostgreSQL database. The user scrolls through twenty-three (23) pages of the report generated from the Bosch CDR Software.

47.   The video and further investigation reveals that the following files of interest were present on the Laptop as of April 3, 2018, but are no longer available for review:

47.1.   C:\Users\Brian\PycharmProjects\JSON Processor\ReplayFileMaker.py

47.2.   C:\Users\Brian\Documents:[41]

47.2.1.   Bosch CDR.dbc;

---

[41] As shown in files listed in "C:\Users\Brian\Documents" at approximately 1:28 pm in 8-19-22 video.

47.2.2.   CDR.ini;

47.2.3.   Bosch CDR.cfx; and

47.2.4.   CDR.au3. Based on names and extensions, these files, which are no

longer present on the Laptop, appear to have been related to BUSMASTER and

Respondent's "CDR Replay" tool.

Table 3: Files of Interest No Longer Present on the Laptop

1	C:\Users\Brian\PycharmProjects\JSON Processor\ReplayFileMaker.py
2	C:\Users\Brian\Documents\Bosch Cdr.dbc
3	C:\Users\Brian\Documents\CDR.ini
4	C:\Users\Brian\Documents\Bosch CDR.cfx
5	C:\Users\Brian\Documents\CDR.au3

48.   Additionally, production from counsel in the Developer Notes[42] further indicates that

the goal of the "CDR Replay" tool was to "'replay' one of our crash data files into the Bosch

CDR software." And that the "AutoCDR" tool "merely automates running the Bosch CDR

software, using AutoIT."[43] This document appears to contain developer notes on progress

related to the "CDR Replay" tool, and that as of May, 30, 2016 it could extract PID data from a

"trace" and "Play that back into the CDR. It worked, and generated a report. As for security

access, I can now get the CDR to give up its secrets. For Toyota, I have the algorithm figured

out…The security key may change year to year, but that can be worked out."[44] Additionally,

the document indicates that as of June 15, 2016 there was a plan to attempt to "use the Bosch

CDR remotely," when vehicles were not supported by Respondent, record the "trace" for each

new part, and that "would theoretically let us support 100% of vehicles out of the gate."[45] As of

June 29, 2016, a developer for Respondent was:

[W]orking on full automation of the CDR report generation. This involved

---

[42] CSI00437680
[43] CS00437680, p. 1.
[44] CS00437680, p. 2.
[45] CS00437680, p. 3.

running the Bosch software automatically and playing recorded data into it. That's working now within a virtualized environment. The next step here is to poll the server for new reports, so the whole process is automated.[46]

49.   The Developer Notes also corroborate earlier findings and suspicions, stating that the "AutoCDR software was created separately (using AutoIT3)...(and the developer is named Thai and is still available)." Also, "CarCrash[.exe] was a continuation of the software, in an attempt to use with a tablet and the green box in the field...to place a 'telematics' order to my server prior to 'auto-running' the cdr software." This evidence supports a finding that AutoCDR and CarCrash.exe were part of an attempt to automate the running of the Bosch CDR Software.

50.   Further, the "telematics project summary" describes a prototype and goal software of remotely reading PIDs from a vehicle, sending to a server, then having that data "played back into the Bosch lab hardware/software," to create reports, with the further goal to have a system that used a Bluetooth obd dongle, mobile app to collect and send PID data to a server, replay the data into the Bosch CDR Software, then automatically email a crash data report to the end user.[47] The "play back" of the data in the Bosch CDR Software is likely Respondents "CDR Replay" tool. The document additionally includes links to a "CAN hack sniff/trace tool that we used. i.e. an OBDII to USB solution that monitors all CAN frames".[48]

C.   Bosch CDR Software version 21.5.1 was run on the Laptop

51.   There is evidence that the Bosch CDR Software version 21.5.1 was run on the Laptop at least on one occasion, on July 19, 2022, per the crash dump file C:\Users\Brian\AppData\Local\CrashDumps\CDR.EXE.9656. This file is noteworthy for two

---

[46] CS00437680, p. 5.
[47] CS00437680, p. 8.
[48] CS00437680, p. 12.

reasons: first, the application crash for the Bosch CDR Software is for a recent version, version 21.5.1; and second, the "licensed company" is "StreetDelivery," and not CSI. **This crash dump file was likely generated on the Laptop when Bosch Software version 21.5.1 was running on the laptop using the StreetDelivery license file.** This is evidence that Respondent ran Bosch CDR Software using other customer licenses and certificates on the Laptop.[49] It also raises the question as to why CSI would be running Bosch CDR Software on this Laptop using a license from a different company.

> **D.** **The Laptop contained license certificate files for the Bosch CDR Software for companies other than Respondent**

52. Similar to the StreetDelivery license above, the Laptop contained Bosch CDR Software license certificates for several versions of the Bosch CDR Software licensed to third-party companies that were not Respondent, namely Street Delivery and Biologic Forensics. The Laptop contained activation certificate files, "CTF files," that appeared to be for "1 year" terms, located in the directories C:\Users\Brian\Documents\CDR Installation\ and C:\Users\Brian\Documents\CDR Installation\Supplier\, e.g., "CDR21.2_1_1yr_1ED17E72_SD.CTF." The "CDR Installation" folder contained CTF files for CDR versions 21.2; 21.3, 21.4, and 21.5, last modified between October 15, 2021 and April 9, 2022. The "Supplier" folder contained CTF files for CDR versions 19.4, 19.6, 21.0, 21.1, 21.2, 21.3, 21.4, and 21.5 last modified between January 21, 2021 and May 9, 2022.[50]

---

[49] Visible at approximately 12:54 pm in 8-25-2022 Capture Video, the CDR.EXE Application crash was from July 19, 2022.
[50] Seen at approximately 1:16 pm in 8-25-2022 Capture Video; Supplier certs reviewed at approximately 3:47 pm in 8-25-2022 Capture Video.

Table 4: CTF Certificate Files on Laptop for companies other than CSI in the "Supplier" folder

Bosch CDR Software Version	Company listed within the CTF file
21.5	Street Delivery
21.5	BioLogic Forensics
21.4	StreetDelivery
21.3	StreetDelivery
21.2	StreetDelivery
21.2	BioLogic Forensics
21.1	BioLogic Forensics
21.0	BioLogic Forensics
19.6	BioLogic Forensics
19.4	BioLogic Forensics

53. The "CDR Installation" folder included CTF files with the company listed as Collision Sciences, including Bosch CDR Software versions 16.4, 17.10, 19.1, 19.2, 19.3, and 19.4. Curiously, these CTF file versions did not include Bosch CDR Software version 17.7, as shown in the April 3, 2018 "CDR Replay" demonstration video, nor Bosch CDR Software version 17.9, which was run and crashed at least 73 times on the Laptop.[51]

E. CSI used automation tools to automate the running of Bosch CDR Software

54. Certain files present on the Laptop are related to automation, or allowing the operation to run without human intervention, of the Bosch CDR Software. Discussed above briefly as related to the Developer Notes the files, "AutoCDR.au3," "Constants.au3," and "CarCrash.exe" are related to an automation tool, AutoIt, and automating the running of the Bosch CDR Software.[52] They are located within "C:\Users\Brian\Desktop\Desktop\SOURCE CODE\CDR" and "C:\Users\Brian\Desktop\Desktop\SOURCE CODE\CDR and code related\random code files\CDR.zip," and contain scripting logic to automatically execute,

---

[51] As seen at approximately 3:21 pm in the 8-16-22 video.

[52] "AutoIt v3 is a freeware BASIC-like scripting language designed for automating the Windows GUI and general scripting. It uses a combination of simulated keystrokes, mouse movement and window/control manipulation in order to automate tasks in a way not possible or reliable with other languages (e.g., VBScript and SendKeys). AutoIt is also very small, self-contained and will run on all versions of Windows out-of-the-box with no annoying "runtimes" required!" Available at https://www.autoitscript.com/wiki/AutoIt_Introduction.

interact with, and control the Bosch CDR Software application, as well as saving reports using the AutoIt automation tool. AutoCDR.au3 was last modified on June 28, 2016, but the containing CDR.zip was created on the Audit Laptop on May 31, 2021. CarCrash.exe likely required Bosch CDR.EXE to be installed in order to run, based on the Developer Notes, as well as a review of evidence within the CarCrash.exe executable itself, "AutoCDR.au3," "Start CDR.bat," and "Constants.au3" files.[53]

55.   Further, the text of one of these AutoIT scripts, "CDR.au3", has been produced as CS00016681-83. The "CDR.au3" file includes code for automating the Bosch CDR Software, including "opening a new file window in CDR", waiting until CDR is ready, clicking on a button within the Bosch CDR Software to "Read VIN from Vehicle", launching the BUSMASTER tool, sending a series of keyboard key commands that appear to generate a report if a variable "$report" is set to "1", and a different set of keyboard key commands if no report is needed. A sample screenshot of the code performing the described functionality is included below:

---

[53] CarCrash.exe executable itself contains references to "AutoCDR.exe", which is likely the executable file for "AutoCDR.au3" AutoIT script file. The AutoCDR.au3 script file required the Bosch Software tool "CDR.EXE" executable to be installed to run, per "AutoCDR.au3" and "Constants.au3", from approximately 2:30 pm to 2:37 pm in the 8-19-22 video. Additionally, references within the "CarCrash.exe" executable itself point to C:\Users\thaingo84\Dropbox\oDesk\AutoIT-collisionsciences\CarCrash\CarCrash\obj\Debug\CarCrash.pdb; pdb files are generally created as program debug files by visual studio during application development. Lastly, CarCrash.exe is located next to "AutoCDR.exe" in the "Application" directory within "AutoCDR.zip", inside C:\Users\Brian\Desktop\Desktop\SOURCE CODE\CDR and code related\random code files\CDR.zip archive, as shown at 12:48 pm in the 8-18-22 video.

Image 10: Sample Screenshot from "CDR.au3" AutoIT Automation Script

```
;Send("^n") ;This opens a new title window in CDR

;Sleep(500)

;Send("c")
;Send("{ENTER}")

;Sleep(500)

; Now the CDR is ready to read the VIN. Click on the "Read VIN from Vehicle" button.

;ControlClick("Bosch Crash Data Retrieval Tool - [Vehicle identification]","","[ID:5]")

;WinWaitClose("Bosch Crash Data Retrieval Tool - [Vehicle identification]")

;Sleep(200)

WinActivate("BUSMASTER","")

Sleep(200)

Send("c")

Sleep(200)

WinActivate("Bosch Crash Data Retrieval Tool","")

Sleep(200)

If $report == 1 Then
Send("{LALT}")
Send("{RIGHT}")
Send("{RIGHT}")
Send("{RIGHT}")
Send("{DOWN}")
Send("{ENTER}")
Else
Send("{LALT}")
Send("{RIGHT}")
Send("{RIGHT}")
```

CS00016682

56.   The "CDR.au3" file also includes some branching to wait based on the output of the

Bosch CDR Software, such as if certain messages like "Pass Two" or "ECU Part Number Not

Support" are presented by the Bosch CDR Software. On how the controls of the keyboard were

automated to save out a Bosch CDR report, with branching logic for different error messages

and scenarios from the Bosch CDR Software on how to end the AutoIT script.

F. There is evidence on the Laptop related to potential reverse engineering of the Bosch Software

57.   First and foremost, Respondent's "CDR Replay tool" software incorporates the Bosch CDR Software, as well as the "eavesdropping" software BUSMASTER, which appears to be used to attempt to mock up a connection to a vehicle, and "sniff" or overhear the messages sent to and from the Bosch CDR Software, as described above.

58.   In addition to these items, memory dumps from the PyCharm development environment for Python on the Laptop ("heap dump" files) contain references to files that no longer exist on the Laptop. These deleted files also indicate that reverse engineering efforts may have been occurring on the Laptop. Memory heap dumps indicate that files listed within the dump were loaded into the memory for PyCharm application when the memory dump was created.[54] For example, one memory dump contains references to "C:\Users\Brian\Documents\Bosch CDR.ini"; "Bosch CDR.dbc"; "Bosch CDR.cfx"; "CDR.au3"; and "vehicle.py."[55] These files are related to the BUSMASTER and the "CDR Replay tool":

   58.1.   Vehicle.py is a code file that is part of CSI's cdrservice application.[56]

   58.2.   The "Bosch CDR.dbc" file may have been a Bosch database itself as related to the Bosch CDR Software, however the file is no longer present on the Laptop for review.

---

[54] "A heap dump is a snapshot of all the objects in the Java Virtual Machine (JVM) heap at a certain point in time... by examining the heap you can locate where objects are created and find the references to those objects in the source." More information on heap dumps is available at https://docs.oracle.com/javase/8/docs/technotes/guides/visualvm/heapdump.html.
[55] Located within "heapDump-pycharm-1623080586417.hprof" located at "C:\Users\Brian\AppData\Local\JetBrains\PyCharmCE2021.1\tmp\hprof-temp\" as shown at approximately 2:31 pm in 8-18-2022 Capture Video.
[56] As shown at approximately 2:32 pm in the 8-18-22 video.

From Google searches, "DBC" files may be CAN Database files. It is certainly unusual that a file named "Bosch CDR.dbc" was present at C:\Users\Brian\Documents\ in the past.[57] This file is no longer present in the "C:\Users\Brian\Documents" folder.[58]

58.3.   "Bosch CDR.cfx" is related to the "CDR Replay tool" from the April 3, 2018 "CDR Replay.wmv" video; it appears to be a configuration file for the BUSMASTER application that was loaded for the demonstration of the "CDR Replay tool", as seen in the April 3, 2018 "CDR Replay.wmv" demonstration video. This configuration file is no longer present on the Laptop.[59]

58.4.   "Bosch CDR.ini" is an initialization file for running the Bosch CDR Software. This file is no longer present in the "C:\Users\Brian\Documents" folder.[60]

58.5.   "CDR.au3" was likely an automation script created using AutoIt; the Laptop contains evidence indicating that AutoIt was likely used to automate running of the Bosch CDR Software. This file is no longer present in the "C:\Users\Brian\Documents" folder.[61]

59.   In addition to evidence of potential reverse engineering in the memory dumps from PyCharm, the Laptop contains thousands of Bosch CDR reports generated by using the Bosch CDR Software. Presence of thousands of Bosch CDR reports containing the data exported from the Bosch CDR Software could be indicative of attempts to reverse engineer the messages and data used to generate the Bosch CDR reports.

60.   The Laptop also contains what appears to be data that may have been exported from databases into Microsoft Excel, located in "C:\Users\Brian\Downloads\Toyota ETI\".[62] The

---

[57] As shown at approximately 1:37 pm in 8-19-22 video; search hit found in
C:\Users\Brian\.PyCharmCE2019.3\system\LocalHistory\changes.storageData.
[58] As shown in files listed in "C:\Users\Brian\Documents" at approximately 1:28 pm in 8-19-22 video.
[59] As shown in files listed in "C:\Users\Brian\Documents" at approximately 1:28 pm in 8-19-22 video.
[60] As shown in files listed in "C:\Users\Brian\Documents" at approximately 1:28 pm in 8-19-22 video.
[61] As shown in files listed in "C:\Users\Brian\Documents" at approximately 1:28 pm in 8-19-22 video.
[62] As shown at approximately 11:03 am and 11:08 am in the 8-19-22 video.

files, "DB_Vehicle_V153_NA_d.xlsx" and "DB_Vehicle_V151_NA_c.xlsx", appear to be databases or database dumps from ten to twenty tables containing a host of ECU data related to Lexus and Toyota vehicles. The origin of this data is unclear.

61.   The Laptop contains Toyota parts listings located in C:\Users\Brian\Desktop\Desktop\Temp Toyota List.txt and C:\Users\Brian\Desktop\Desktop\Python Files\PartNumber.txt. These part listings contain the part number prefix "89170." It is my understanding from Bill Rose, a Senior Product Manager at Bosch, that this part number prefix is not part of Toyota's part number, but rather it is a prefix that was added by Bosch for use in its Bosch CDR Software, and is not used elsewhere. It is unclear where these part numbers with the Bosch prefix could have originated, if not extracted or reverse engineered through use of the Bosch CDR Software.[63]

62.   Additionally, these part numbers found on the Laptop include numbers for Bosch hardware supported only in Japan and Australia, not the US or Canada: "75010, 75020, 75030, 75040," and "0W310, 0W320," respectively.[64] The "Temp Toyota List.txt" was last modified on 2019-05-04. Text file appears to be mostly duplicated in C:\Users\Brian\Desktop\Desktop\Python Files\PartNumber.txt; also located next to "ToyotaPartNumberParser.py," The Python code file from that directory, "ToyotaPartNumberParser.py," opens a file named "Toyota.txt," parses it to clean up the text data, then outputs "PartNumber.txt." This is possibly indicative of reverse engineering using Bosch hardware and software to attempt to parse and process the "ToyotaPartNumber."[65]

---

[63] As seen at approximately 12:33 pm in the 8-19-22 video.

[64] As seen at approximately 12:36 pm in the 8-19-22 video for 0W310 and 0W320 relevant Australia part numbers; 75010, 75020, 75030, 75040 for Japan at 12:40 pm in 8-19-22 video.

[65] Parse can be defined as, "to determine the syntactic structure of a language unit by decomposing it into more elementary subunits and establishing the relationships among the subunits," per (ISO/IEC/IEEE 24765:2017 Systems and software engineering-Vocabulary), as defined on pascal.computer.org. Parsing is essentially deconstructing data to make it more useful and/or understandable.

63.   Files found on the Laptop with the ".org" extension located within the

"C:\Users\Brian\Desktop\Desktop\SOURCE CODE\CDR and code related\cars\" directory

appear to be notes on progress of development and planning for Collision Sciences own Crash

Data tool, including:

63.1.   "support.org" contains notes about vehicles, including this note about "Toyota",

stating, "Seems to just be one system, but different security keys, easy to crack";[66]

63.2.   "Plan.org" appears to be a "CDR plan" for development at a high level for

Collision Sciences, last modified June 6, 2016. The "CDR plan" includes tasks for

"read unsecured PIDs from vehicle," "security unlock," "read secured PIDs from

vehicle," "Create PID file", and "replay PIDs into CDR -> new tool (uds-replay)." The

"uds-replay" tool may have been a precursor to the "CDR Replay" tool demonstrated

in the April 3, 2018 CDR Replay video. This file also includes a plan for "Remote

Diagnostics" describing a remote vehicle client and a centralized server.[67]

---

[66] As shown at approximately 12:50 pm in 8-19-22 video.
[67] As shown at approximately 12:51 pm in the 8-19-22 video.

Image 11: Screenshot of top 26 lines of "plan.org"

63.3.   "#seedkey.org#" appears to be notes attempting to bypass and/or reverse engineer the seed and key security for a Toyota 2012 Prius; last modified on May 30, 2016.[68] Bosch Toyota 89170 part number prefixes are also present in #seedkey.org# file.

---

[68] As shown at approximately 12:54 pm in the 8-19-22 video.

Image 12: Screenshot of the first 26 lines of #seedkey.org#

63.4.   "stuff.org" from Eric Dropbox, similar to other .org files, appears to be a listing of goals; lists "Remote CDR" on what appears to be an aspirational task list, as shown below.

Image 13: Sample text from "stuff.org" file
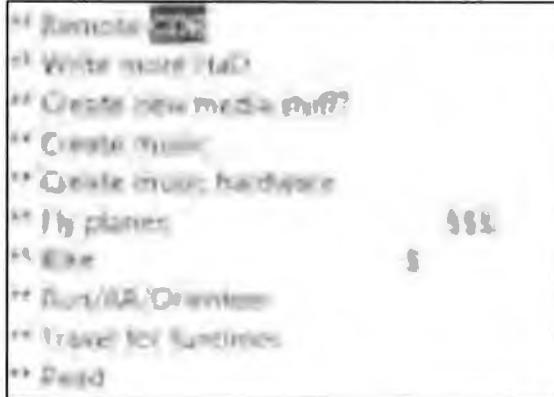
63.5.   "Research.org" file in "C:\Users\Brian\Desktop\Desktop\SOURCE CODE\CDR

and code related\cars\" folder contains what appears to be notes on reverse engineering

testing and progress, including Bosch Toyota part prefix numbers.[69]

Image 14: Sample text from "research.org" file
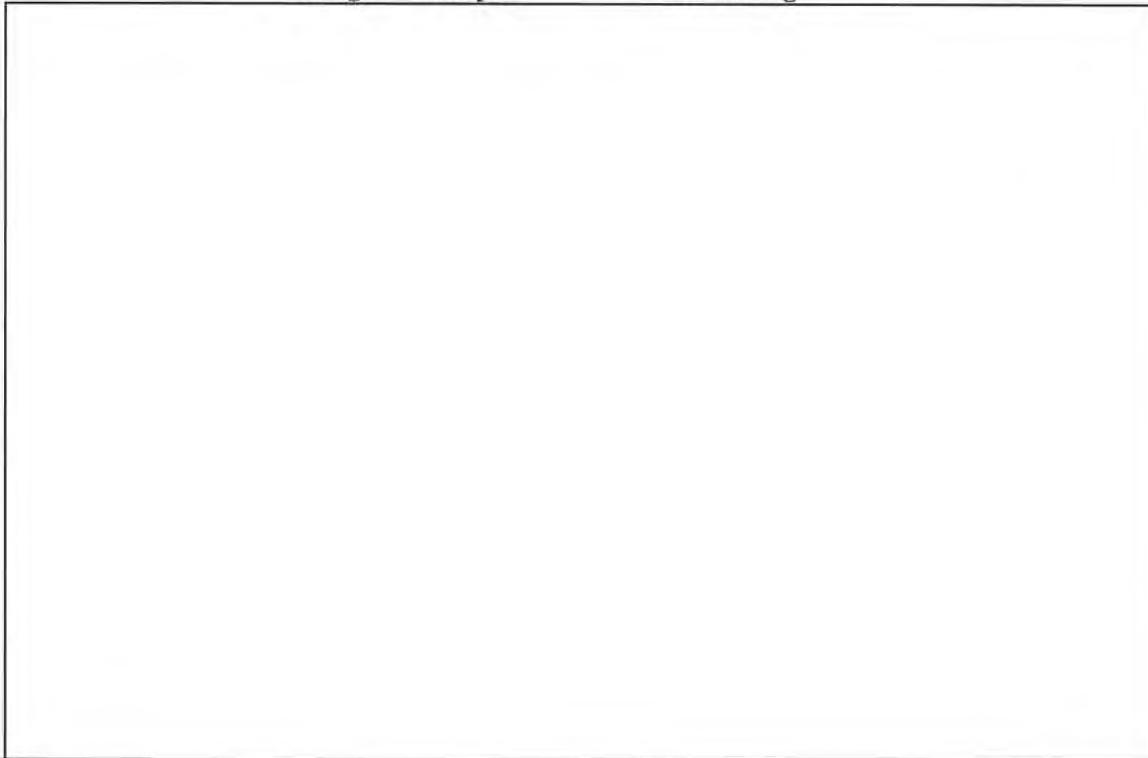
63.6.   Additionally, files in this "C:\Users\Brian\Desktop\Desktop\SOURCE

---

[69] As seen at approximately 12:45 pm in the 8-19-22 video.

CODE\CDR and code related\cars\" directory appear to have been moved from a Mac due to "._[name]" files also present, which is consistent with the theory that an earlier computer used for developing the "CDR Replay" tool was an Intel-based Macintosh computer running the Bootcamp software, as referenced in Item 6(a) within Respondent's Explanations Pursuant to Paragraph 8 of Audit Protocol Order. Files on the Laptop further indicate that research.org and seedkey.org were once stored on a Dropbox account for user "eric" within an emacs.d.zip file.[70]

63.7.   "C:\Users\Brian\Desktop\Notes.txt" also appears to contain notes on reverse engineering progress related to hex data and Toyota seed-and-key security. The "Notes.txt" file was last modified on June 7, 2022.[71]

Image 15: Sample 1 of text within the "Notes.txt" file

Image 16: Sample 2 of text within the "Notes.txt" file

---

[70] As seen at approximately 12:57 pm in the 8-19-22 video.
[71] As seen at approximately 2:15 pm in the 8-19-22 video.

G. The CSI CrashScan repositories made available to me did not include the "cdr-replay-controller".

64.   It appears as though an additional source code repository on Bitbucket, "Collisionsciencecdr/cdr-replay-controller," exists or existed, but was not made available to me as part of the Audit Systems.[72] Based on the findings on the Laptop related to CSI's "CDR Replay" tool, it seems that the data contained within this repository, should it exist, would likely be relevant to the questions and issues I have been asked to investigate, insofar as the "cdr-replay-controller" repository pertains to Respondent's "CDR Replay" tool.

H. The "CS_Production" database hosted on Amazon Web Services appears to be based upon and expanded from the "cdr" database shown in the April 3, 2018 CDR Replay tool demonstration video used in conjunction with the CSI CDR Replay tool.

65.   CSI has hosted a database titled "CS_Production" on its Amazon Web Services ("AWS") platform.[73] This "CS_Production" database appears to be based upon the "cdr" database shown in the April 3, 2018 CDR Replay tool demonstration video, but expanded to include many more tables in the "CS_Production" database. Additionally, the unique identifier value for a given row in the "cdr" table appears to have been updated to use what appears to be a unique hash value[74] in the "CS_Production" database instead of a simple number value in the older, April 3, 2018 database. Data shown in the "cdr_data" table appears to be the same type of information generally stored in the current "cdr_data" table. The current "CS_Production" database is much more expansive, however, with many more tables (69 total) and data. This is

---

[72] *See* CS00444978 and CS00013052.

[73] There are also several mirrors, clones, and replication of this database in the CSI AWS Environment.

[74] "A hash value is a numeric value of a fixed length that uniquely identifies data. Hash values represent large amounts of data as much smaller numeric values, so they are used with digital signatures. You can sign a hash value more efficiently than signing the larger value. Hash values are also useful for verifying the integrity of data sent through insecure channels. The hash value of received data can be compared to the hash value of data as it was sent to determine whether the data was altered," available at https://learn.microsoft.com/en-us/dotnet/standard/security/ensuring-data-integrity-with-hash-codes.

not surprising, as CSI has had many years to continue development of its Crash Scan product

over the more than five years from April 3, 2018 until the present day.

66.   The Amazon RDS instance for "CA-Central," contained three databases:

66.1.   Cs-development, created on September 12, 2019 13:59 pm;[75]

66.2.   Cs-production, created on September 12, 2019 13:55 pm;[76] and

66.3.   Cs-production-read-replica, created on April 27, 2022 1:20 pm.[77]

67.   The focus of my analysis of the database was on the "Cs-production-read-replica," as

that database should simply be a read-only copy of the production database.[78] The database for

the "Cs-production-read-replica" was named "CSProductionDB", and contained 69 tables.[79]

These database table names are listed below.

Table 5: Tables in "CSProductionDB" Replica Database

Access_record	Admin_ip	Aggregate_data
Api_call_record	App_config	Archive_data
Ascm_support	Carbon_copy	Carmd
Cdr_data	Cdr_report	Claim_info
Client_representative	Client_representative_app_config	Client_representative_product_fee
Client_representative_subscription	Condition	Country_detail
Cron_job_control	Cs_admin	Domain
Domain_carbon_copy	Domain_email_rule	Duck_creek_claim_info
Edr_support	Email_queue	Email_rule
Email_task	Emissions_dtc	Exchange_rate
Failed_scan	Frequency	Group
Group_carbon_copy	Hardware_purchase	Invoice_data
Likely_unsupported	Management_document	Management_report
Organization	Organization_detail	Prepayment
Product	Product_credit	Product_fee
Purchase	Purchase_authorization	Report_config
Report_fee	Report_purchase	Report_purchase_backup
Report_review	Role	Scan_error_log
Scan_meta_data	Secondary_did	Subscription
Toyota_module	User	User_group
User_invite	User_role	Vehicle_group_exception
Vehicle_model	Versions	Vin
Vin_audit_history	Vin_decoding	Xero_token

---

[75] As seen at approximately 3:15 pm in the 12-7-2022 Capture Video.

[76] As seen at approximately 3:12 pm in the 12-7-2022 Capture Video.

[77] As seen at approximately 3:09 pm in the 12-7-2022 Capture Video.

[78] The database was located at cs-production-resd-replica.clxcwbawcc3f.ca-central-1.rds.amazonaws.com. I connected to the replica database named "CSProductionDB" using pgAdmin and the credentials I was provided.

[79] As seen at approximately 3:45 pm in the 12-12-2022 Capture Video.

68.   The "CSProductionDB"also contained a table named "cdr_data." This table contained

eighteen (18) columns, named below:

Table 6: Columns in the "cdr_data" table in the "CSProductionDB" Replica Database

id	date_created	date_modified
vin	year	make
model	cdr_data	report_created
email	scan_result	crash_type
delta_v	process_name	error_flag
user_id	organization_id	report_number

69.   It is not clear if all of the columns present in the "cdr_data" table in April 3, 2018 were

visible in the April 3, 2018 "CDR Replay" demo video, however each of the columns that are

visible appear to have analogues in the newer "cdr_data" table. Ten (10) columns are visible,

nine (9) of which appear to have direct analogues in the newer "cdr_data" table, and one (1),

the "vehicle" column which held the year, make and model, appears to have been split out into

three separate columns in the newer "cdr_data" table, with one column each for "year",

"make", and "model", respectively. This correlation is shown in the table and screenshots

below:

Table 7: Visible columns in "cdr_data" from April 3, 2018 vs. current "cdr_data" columns

"cdr_data" (2018)		"cdr_data" (2022)
Id		Id
Date_created		Date_created
Date_modified		Date_modified
Vin		Vin
		Year
Vehicle		Make
		Model
Cdr_data		Cdr_data
Report_created		Report_created
Email		Email
Code		Scan_result
type		Crash_type
Delta_v		Delta_v

Image 17: Screenshot comparison of "cdr_data" tables

70.   Additionally, the unique identifier column "ID" appears to have been updated from a simple numeric value as of the April 3, 2018 video (as shown in the screenshot above to have a value of "1056"), to a more complicated hexadecimal value, as shown in the screenshot below in the red box.

Image 18: Example Hexadecimal "ID" Values in new cdr_data table

71.   I ran queries against the "cdr_data" database table. My first query was to learn about

the size of data within the newer "cdr_data" table. With my first query, I found that the newer

"cdr_data" table contained 86,067 rows, as compared to the table as of April 3, 2018, which

contained only 1057 rows, and that new database appears to contain the same type of JSON

information in the "cdr_data" column as was displayed in the 2018 video.[80] Queries (2)

through (5) were attempts to locate the data for the 2012 Toyota Camry processed in the April

---

[80] As seen at approximately 4:04 pm in the 12-12-2022 Capture Video.

3, 2018 "CDR Replay" demo video, where I learned that the earliest data records are from January, 2018, and that the 2012 Toyota Camry record did not appear to be present in the new "cdr_data" table. The queries I ran and the results are included in table form below:

Table 8: SQL Queries and Findings from AWS

#	Query	Finding
1	Select * from cdr_data[81]	86,067 rows returned as compared to 1057 unique ids in 4/3/2018 video.
2	Select * from cdr_data where ID == 1056[82]	ID == 1056 (2012 Toyota Camry used in Replay video). Failed query as ID appears to be a generated hex key now instead of a simple number for uuid.
3	Select * from cdr_data where year = 2012 and make = 'Toyota' and model = 'Camry'[83]	2012 Toyota Camry from Replay Video not found
4	Select * from cdr_data where year = 2012 and make = 'Toyota' and model = 'Camry' order by date_created asc[84]	Oldest record for 2012 Toyota Camry is from 2019-08-14-14:06, much later than the April 3, 2018 "CDR Replay" demonstration video
5	Select * from cdr_data order by date_created asc[85]	The earliest creation date in the cdr_data table is 2018-01-02 17:51; so it is surprising that the data used in the "CDR Replay.wmv" from April 3, 2018 is not present within the data. There is a "Date created" gap between 2018-01-04 and 2018-04-20 of several months, which would encompass the timeframe of the April 3, 2018 "CDR Replay.wmv" demonstration video's creation, which may also explain the discrepancy.

72.   I also ran queries against the database for the "edr_support" table. The "edr_support" table included columns for "cdr_900", "cdr_canplus", and "source". The fifteen (15) columns in the edr_support table are shown below:

---

[81] As seen at approximately 3:53 pm in the 12-12-2022 Capture Video.
[82] As seen at approximately 3:59 pm in the 12-12-2022 Capture Video.
[83] As seen at approximately 3:59 pm in the 12-12-2022 Capture Video.
[84] As seen at approximately 4:01 pm in the 12-12-2022 Capture Video.
[85] As seen from approximately 4:02 to 4:07 pm in the 12-12-2022 Capture Video.

Image 19: Columns in the "edr_support" Table
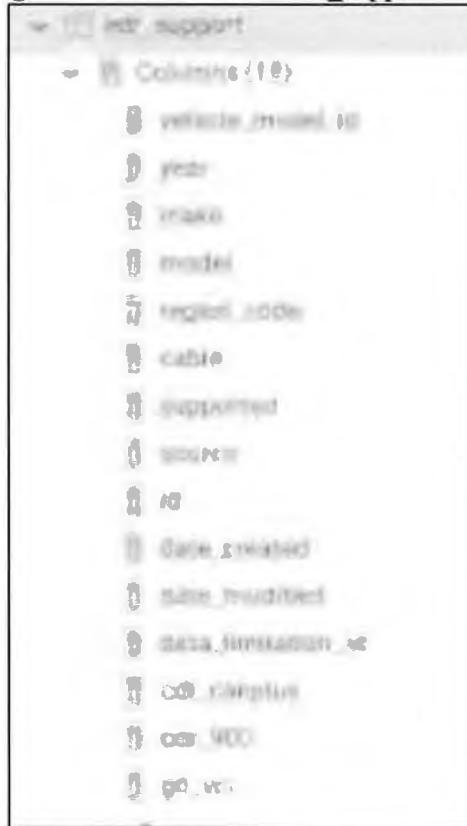
73.   Query (6) was to perform a general search of the data within the "edr_support" table. This query revealed that the "Source" column contained entries for "CDR v19.0" and "CDR v19.4", among other CDR versions, as well as a "Source" listed as "Manual". Query (7) was to isolate records where the value in the "Source" column started with the text "CDR". Queries (8) and (9) were to further refine on the CDR verisions, and to determine each distinct CDR version listed within the "Source" column in the "edr_support" table.

74.   The queries I ran against the "edr_support" table and the results are included in table form below:

Table 9: SQL Queries and Findings from AWS (cont.)

#	Query (cont.)	Finding
6	Select * from edr_support order	12,520 total records; earliest date_created April 14, 2020; earliest modified July 9, 2021. "Source" column indicated "CDR v19.0" and "CDR v19.4"

|		
	by date_created asc[86]	
7	Select * from edr_support where source like 'CDR%' order by date_created asc[87]	8,410 where source like 'CDR%'. Source data listed as "CDR 19.0" and "19.4" in this data table; "date_created" of April 24, 2020 for 19.0; May 8, 2020 for CDR v. 19.4, and "date_modified" at least as late as July 10, 2021.
8	Select distinct source from edr_support where source like 'CDR%'[88]	revealed 17 versions of Bosch CDR Software listed, refined in the next query
9	Select distinct source from edr_support where source like 'CDR%' order by source asc[89]	Query revealed "sources" listing CDR versions: 18.0; 18.0 – ZA; 19.0; 19.0 – ZA; 19.1; 19.2; 19.3; 19.4; 19.5; 19.6; 21.0; 21.1; 21.2; 21.3; 21.4; 21.5; 23.0

---

[86] As seen at approximately 4:10 pm in the 12-12-2022 Capture Video.
[87] As seen at approximately 4:14 pm in the 12-12-2022 Capture Video.
[88] As seen at approximately 4:16 pm in the 12-12-2022 Capture Video.
[89] As seen at approximately 4:16 pm in the 12-12-2022 Capture Video.

Respectfully submitted,

July 31, 2023.
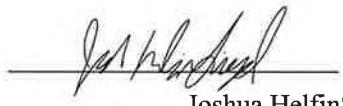
Joshua HelfinSiegel

ATTACHMENT 1
CV OF JOSHUA HELFINSIEGEL

Mr. HelfinSiegel has more than sixteen years of experience as an IT professional, including Systems Administration, Security, and Technical Support experience. Since 2011, he has worked as a litigation consultant for DisputeSoft. He is responsible for all of the environments, data, and systems in use at DisputeSoft, including data security and Domain management. He has extensive experience in dealing with the complex issues surrounding large software implementation failures, copyright infringement and trade secret misappropriation, and digital forensics. He has worked on a variety of software failure, patent, copyright, and digital forensics cases and has performed numerous analyses of: Software Requirements, Design, Development, Testing, Defects, Software Quality, and Project Schedules. He has performed the AFC test, as well as analyzed various code sets for evidence of copying to support infringement or misappropriation claims. He has analyzed digital forensic evidence, including searches for evidence inappropriate copying of confidential information, and evidence of deletion of said information. He specializes in failed software implementations; intellectual property; databases and data analytics; computer networking, hardware, and infrastructure; and digital forensics. Mr. HelfinSiegel is an EnCase certified forensic examiner.

Prior to starting at DisputeSoft, Mr. HelfinSiegel worked as the IT Manager for a property management company, TM Associates Management. There he served as the sole Systems Administrator, IT Director, and IT Support point of contact for a company of over 150 distributed locations and over 100 remote and 20 centralized employees. He was responsible for all of the environments, data, and systems in use at the company, including data security and Domain management. Responsibilities also included the installation and maintenance of several antivirus and malware protection software, malware and virus removal for over 100 remote sites, protection of an in-house network against internet threats and vulnerabilities, and hardening servers against potential points of attack.

DISPUTESOFT EXPERIENCE

Copyright Infringement and Trade Secret Misappropriation Disputes

TruLogic v. GE Aviation (2023)
In this intellectual property case involving alleged breach of End User License Agreement ("EULA") related to creation of derivative works in the aviation industry, assisted counsel with analyzing source code and drafted an expert report and supplement to respond to opposing expert opinions related to Interactive Electronic Technical Manuals ("IETMs"), and to address quality and quantity of alleged copying. Anticipate providing deposition testimony and testifying at trial. Ongoing engagement.

Benchmark Technologies, Inc. v. Yuqiang Tu et al (2022)
In this alleged trade secret misappropriation case in the optical lithography industry, analyzed sets of source code for evidence of copying and misappropriation of trade secrets surrounding highly technical aspects of the lithography as embodied by source code. Ongoing engagement.

Trent P. Fisher Enters. v. SAS Automation, LLC (2022)
In this alleged copyright infringement and trade secret misappropriation case in the manufacturing and robotics industry, assisted counsel with responding to Plaintiff's expert report opinions, as well as in understanding the viral nature of the open-source GNU General Public License, Version 3 as applied to works at issue in this matter. Ongoing engagement.

[party names withheld] (2022 AAA Arbitration)
In this alleged infringement and breach of contract matter, assisted counsel in a computer and source code forensic audit, reviewing keywords search results and source code for indicia of reverse engineering and/or improper use of Plaintiff software. Filed Declaration to support counsel's motion to compel. Ongoing engagement.

Covetrus, Inc. and Veterinary Data Services, Inc. v. Actian Corporation (2022)
In this alleged copyright infringement and license violation matter in the veterinary industry, reviewed and responded to opinions in Plaintiff's expert report regarding copyrights; performed forensic computer analysis and cloud virtual system audits to search for indicia of the presence of installed software to compare against the allowed number of installs per software license. Provided expert report and deposition testimony. Ongoing engagement.

[party names withheld] (2021)
In this copyright infringement, trade secret, and patent infringement case in the banking industry, analyzed deposited copyright material as related to alleged trade secrets for evidence of disclosure of trade secrets in the registered work, as well as analysis related to whether the version of code contained in the copyright material appeared to be a true and accurate copy of what was represented as the registered work. Additionally, decompiled and disassembled android and iOS code for comparison against alleged trade secrets. Ongoing engagement.

Ubiquiti Networks v. Cambium Networks, et al. (2020)
In this copyright infringement and breach of contract case, engaged due to familiarity and expertise regarding source code analysis and the GPL license. Analyzed different registered versions of software programs for presence of open-source code protected by the GPL license to support counsel arguments regarding copyright infringement, breach of contract, and to determine which source code elements would be protectable but not affected by the copyleft nature of the GPL license. Conducted on-site source code review of opposing party's code to search for evidence of copying of the registered works.

Bethesda Softworks, LLC v. Behaviour Interactive, Inc. et al (2018)
In a copyright infringement and breach of contract case involving allegations that a software vendor had misappropriated plaintiff's source code in the development of a mobile video game built in C# and using the Unity engine, assisted counsel with restoring perforce source code repositories, extracting relevant source code and design documents, analyzing the source code and design documents for evidence of copying, and filtering of third-party and non-protectable portions of the code.

[party names withheld] (2017)
In this intellectual property case involving alleged theft of trade secrets related to databases and source code in the medical healthcare data industry, assisted counsel with drafting production requests, interrogatories, and directing production requests toward acquiring the materials needed for expert analysis with regard to trade secret misappropriation allegations in this case. Assisted counsel by drafting Declarations and Expert report regarding data flows within a legacy system to explain to the court how trade secret misappropriation would have occurred based on how the system was configured. Anticipating database analysis, audit log analysis, source code repository analysis, among other analyses after production has been made available. Ongoing engagement.

Arkeyo v. Cummins (2017)

In this intellectual property case, assisted counsel with analysis of source code, dlls, and compiled code to determine if trade secrets had been unintentionally disclosed. Provided a declaration to counsel detailing the importance of the Defendant producing its source code repository so that code and development could be reviewed and analyzed for evidence of copying from the Arkeyo software.

Atlantic Technology Enterprises, Inc. v. Lincoln Park Savings Bank & Abacus I.T. Inc. (2017)

In this intellectual property case, reviewed the document production and depositions to determine what material the Plaintiff claimed was its proprietary information. Reviewed Windows Server backups for proprietary information in order to determine if misappropriation occurred.

Cobra Systems, Inc. v. Unger et al. (2017)

In this intellectual property case, performed source code comparison and reviewed evidence related to both copyright infringement and trade secret misappropriation claims involving software used to print various labels, such as barcodes. Performed an Abstraction-Filtration-Comparison test between two sets of source code in order to demonstrate that the structure, sequence, and organization of the two software programs was substantially similar. Provided a declaration to counsel detailing the evidence of copying between two sets of source code. The declaration also covered the topic of proper clean room design when creating a new product, in order to avoid misappropriation of trade secrets or other protected intellectual property.

ECIMOS, LLC v. Carrier Corporation (2016)

In this intellectual property case, performed source code comparison and reviewed evidence related to both copyright infringement and trade secret misappropriation claims involving software and hardware used for quality testing air conditioning units, as well as what constituted a software API. Traveled to Collierville, TN to view software and hardware in action at the manufacturing plant. Provided a written Declaration to court regarding the difficulty of copying from the text-based source language to the graphics-based language of the accused product. Provided testimony in person to the same effect on September 1, 2017 at a Preliminary Injunction Hearing. Provided a written Declaration signed October 13, 2017 in support of Carrier's memorandum in opposition to Plaintiff's motion to reopen proof, detailing material issues with the opposing expert's report. Provided a written Expert report signed on October 30, 2017 containing affirmative opinions related to clean room design, database comparison and source code analysis, as well as detailing material issues with the opposing expert's report. Provided deposition testimony on January 11, 2018 related to all previously provided written testimony. Provided a Supplemental Expert Report, signed on April 30, 2018 containing affirmative opinions as well as rebutting the newest assertions from opposing expert, and detailing how to apply the Abstraction Filtration Comparison test with regard to the databases at issue in this case. Lastly, provided testimony on June 29, 2018 at jury trial reiterating points made in the Declarations as well as Expert reports and exhibits. Specifically provided testimony focused on database and software issues regarding copyrights and trade secrets, and the Abstraction Filtration Comparison test.

T&S Property Management v. Cinc (2016)
In this intellectual property case, performed source code comparisons between two sets of c-sharp source code and databases to determine if any literal copying had occurred between the programs. Reviewed the code to determine whether one software program was likely derived or reverse-engineered from a competing software program, as reverse-engineering was explicitly prohibited by software license.

QueTel: Consulting (2016)
Analyzed different versions of software programs for evidence of the presence of open-source code protected by the LGPL V 3.0 pursuant to a copyright registration and potential trade secret litigation.

Michael Mohr v. Science and Engineering Services, Inc., et al. (2014)
In this intellectual property case, performed extensive source code comparisons, as well as documentary review, towards determining whether literal copying of source code had occurred outside the scope of a licensing agreement. Interviewed several fact witnesses, drafted interrogatories and deposition questions, performed in-depth analysis related to low-level printer commands and the creation of labels for aircraft. Drafted expert report.

Prosuite Software Limited, et al. v. InfoKey Inc., et al. (2013)
In this intellectual property case performed a class-usage and function-call analysis to determine if any code from one source code set was called in new source code.

Planet Bingo, LLC and Melange Computer Services, Inc. v. VKGS, LLC, d/b/a Video King (2012)
In this software misappropriation case involving casino point-of-sale (POS) gaming software, restored server and client systems, and then performed comparison of plaintiff and defendant software, functionality and documentation to assist in determining whether defendant's software and functionality was substantially similar to and designed using plaintiffs' confidential information.

American Petroleum Institute (2011)
In this copyright infringement case against unknown individuals operating out of China, conducted an investigation to determine the identities of these individuals and determine the extent to which the plaintiff's materials had been pirated. Evaluated websites for potentially infringing content based on PDF standards and sale of copyrighted materials.

Certification Trendz, LTD. v. PassGuide.com et al. (2011)
In this copyright infringement and trademark misappropriation case against unknown individuals operating out of China, conducted an investigation to provide attribution of these individuals and determine the extent to which the plaintiff's materials had been pirated. Used domain tools and other IP address related utilities to find the names and IP addresses of likely culprits of the infringement.

InDyne, Inc. v. Abacus Technology Corporation, et al. (2011)

Performed web server log analysis and environment reconstruction on behalf of the defendant in this trade secret misappropriation case between NASA contractors. Performed forensic keyword search analyses and rebutted opposing expert's claims over infringing content. DisputeSoft demonstrated that the deposited material from the copyright registration of the misappropriated was actually a reconstruction of the original work through an analysis of the source code and the deposit materials on file with the U.S. Copyright Office.

Nexus v. Kroughly, Limesoft et al. (2011)

In this intellectual property case in the emissions monitoring industry, provided an Affidavit and testimony regarding the nature of compressed "tar.gz" files, restoring backups of source code repositories, and xml configuration settings in source code. Testimony provided at trial highlighted the steps and resources available to the Defendants to determine how to restore a backup of a source code repository to a new location. Testimony also discussed how xml configuration files could be used to validate or verify the origin of accompanying source code produced in the case. Forensic analysis work is ongoing as of 2018.

The Studer Group, LLC. v. The Cleveland Clinic Foundation (2011)

In this intellectual property case, worked closely with the client to acquire and differentiate source code repositories of interest for comparison of infringing code. Forensically acquired and compared source code from repositories and rebuilt the source code management system along with the repositories in question. Compared user login and commit date histories between code repositories to show a lack of cross-contamination between projects. Assisted counsel with deposition questions related to the projects and contract in the case.

IT Project Failures

DXC v. Optus (Consulting) (2021)
In this software failure case involving telco service provider software in the telecommunications industry, assisted counsel in understanding the strengths and weakness of their case based on defect ticket reports and other documentary evidence.

Blue Cross Blue Shield of California v. Health Plan Services (2021)
In this ongoing software failure case involving software in the healthcare and medical insurance industry, assisted in reviewing and responding to opposing expert analyses and findings related to industry standards and the ordinary standard of care. Additionally, assisted in providing examples of expert analyses that would ordinarily have been undertaken by opposing experts, but were not.

Cerner v. Fujitsu (2020)
In this alleged software failure case involving health care software, assisted in developing protocols and analyses for expert agreement to be employed during litigation related to defects and software quality.

Bibb County School District v. Dallemand, et al. (2018)
In this software failure case in the education industry, aided both parties as an independent technical expert with a narrow focus on identifying issues surrounding a production dispute. Engaged at the request of the judge in the matter to provide insight and help resolve the technical problems surrounding the software production.

Pennsylvania Department of Labor and Industry v. IBM (2017)
In this ongoing software failure case regarding a large-scale software modernization project, helped direct data preservation efforts, restoration and analysis of key systems, drafted target opinions and Expert report. Managed a large and complex data analysis effort, as well as all internal personnel throughout the process. Directed a robust and wide-ranging source code analysis strategy, ensuring each analysis employed had basis in applicable industry best practices and/or basis in usage during the engagement. Analyses included static code analyses of code quality for complexity, reliability, flexibility and maintainability compared against industry best practices. Analyses also included defect data analyses related to defect potentials, defect removal efficiency, and defect density against industry best practices. Analysis of test data included analysis of unit test script quality, and rate of test execution compared against rate of test exits to estimate a reasonable project completion date. Additionally, analyzed requirements, design, project management and project schedule delay, as well as verified and validated 3rd party reports and assessments during the project. Drafted expert report; reviewed and responded to multiple expert rebuttal reports in drafted surrebuttal report. Settled 2021.

Acumen v. ADS (2016)
In this software failure case regarding the modernization of a Configure-Price-Quote (CPQ) system, analyzed performance benchmarking data, statements of work, software and system requirements, emails and service contracts to determine if the system as delivered met or exceeded the performance requirements as represented by Acumen. Reviewed procedures followed by Acumen in regards to due diligence in vendor selection in the consulting process with ADS.

Federal Signal Technologies, LLC v. Texas Department of Transportation (2014)
In an administrative hearing regarding highway tolling system contract that was terminated for convenience, assisted in a percent-complete analysis of various deliverables specified in the contract. Linked the hardware architecture diagrams to purchased items based on invoices, emails, and the documentary record.

Mary Rutan Hospital v. NextGen Healthcare Information Systems, LLC (2014)
In this software failure case regarding a failed implementation of hospital management software, performed data analysis of defect data towards determining if contractual agreements for support had been met. Interviewed several fact witnesses, drafted interrogatories and deposition questions, reviewed production environment to perform validation testing. Restored ticketing systems for review and analysis. Drafted expert report.

AMC Technology, L.L.C. v. Cisco Systems, Inc. (2013)
In a breach of contract case involving software for connecting call center systems to third-party CRM software, reviewed documents, testimony, and source code to reach opinions regarding how effectively the defendants conveyed information to the plaintiff in a timely, accurate manner in adherence to standard industry practice. Analyzed a list of purported defects identified during performance testing to determine how many issues, if any, would have had a material impact on the defendant's ability to ship the software to customers. Drafted expert report.

Arc-Com Fabrics, Inc. v. Third Wave Business Systems, LLC (2013)
In a software project failure case involving the deployment of an SAP Business One system for use by a textiles manufacturer, assisted in drafting an expert report opining on issues of system instability, slow system performance, poor source code quality, and deviations from industry standard practices. Performed reconstruction of production system environments for validation testing.

CedarCrestone, Inc. v. Affiliated Computer Services, LLC, et al. (2013)
In a software project failure case involving a failed PeopleSoft upgrade, conducted data analyses of defects recorded in HPQC to determine if material defects in the PeopleSoft software developed by the plaintiff prohibited the project from reaching go-live on time. Performed analysis of defects found in later phases of testing that should not have passed initial Unit testing, had proper testing been performed.

American Orthodontics Corporation v. Epicor Software Corporation (2011)
In this software failure case performed reconstruction of the Epicor ordering system, database environment, and web portal. Assisted in developing a script to simulate large volume orders, then used the script to perform functional testing of said system to prove that orders were delayed and even lost by the software. Performed load-testing analysis to rebut claims that the problems were due to insufficient hardware.

Deluca Enterprises, Inc., et al. v. SAP America Inc., et al. (2011)
In a case alleging overselling and under delivering ERP software, conducted an analysis to determine degrees of similarity between two sets of ARIS business process models based on representations of an SAP integrator that allegedly had reference models applicable to 80 percent of their client's business processes. Rebuilt tape library system and catalog in order to review, analyze and restore relevant data from backup tapes.

Toronto Community Housing Corporation v. Information Systems and Services, Inc. (2011)
In this arbitration brought by a social housing authority against a software vendor before the American Arbitration Association, reconstructed the application environment and systems required for the extensive functional testing needed for this case. Developed and employed a functional testing matrix based on the project's contract and functional specifications, performed functional validation testing, and assisted in the preparation of expert report and hearing materials to establish that the defendant delivered software containing material defects and misrepresented its software's state of readiness during procurement.

GC Services Limited Partnership v. Ontario Systems, LLC, et al. (2010)
In this software project failure case, traveled to Houston, TX and performed forensic acquisition of data as well as extensive analysis and reconstruction of systems from the forensically acquired databases and images. Reconstructed the software environment from the ground up to perform functional testing of the claims in the original pleadings. Reviewed the underlying system architecture and assisted in the preparation of a report evidencing spoliation of the system by the plaintiff. Rebutted allegations of system instability and poor project management through extensive review of case documentation, deposition testimony, and project management standards.

Hudec Dental Associates, Inc. v. Multimedia Dental Systems, Inc. (2010)
Performed extensive analysis of system and audit logs between Dental Practice Management Systems at issue in a software failure case. DisputeSoft demonstrated that the software was materially defective, failed to conform to agreed-upon specifications, did not include promised functionality, contained significant security vulnerabilities that rendered it non-compliant with HIPAA privacy requirements, and was not incapable of supporting the business operations for which it was acquired. Rebutted allegations of ongoing system use past the date of contract termination through extensive audit log analysis and system testing.

Software Patent Infringement Disputes

Wapp Tech Limited Partnership Et Al v. Wells Fargo Bank, N.A. (Consulting) (2022)

In this alleged patent infringement case, assisted counsel by researching the Android device emulator from Android studio in understanding how it works by analyzing its underlying technologies, hardware, software, and networking emulation. Additionally, researched the potential origin of network latency and speed default values available in code as related to the pending patent litigation.

[Party names withheld] (Consulting) (2021)
In this potential patent infringement case in the medical device industry, reverse-engineered and analyzed reverse-engineered operating system for presence or absence of features from the patents at issue in the instant matter. Attempted to access and analyze hardware for presence of patented methods via TCP, UDP, and hardware debugging interfaces, such as JTAG, UART, and SWD.

Uniloc USA, Inc., et al. v. Activision Blizzard, Inc. (2013)
In this patent infringement case, installed, tested activation protocols, captured packet and web traffic for several different versions of antivirus and antimalware software. Assisted in the installation and testing on several different windows platforms in order to verify the process used for software activation.

Apple v. HTC Corporation (2010)
In this smartphone patent infringement litigation before the International Trade Commission, reviewed source code for mobile and desktop operating systems related to the patents at issue. Supported invalidity, non-infringement and lack of domestic industry contentions through research, code review and claim charts. Served as a consulting expert.

9

Computer Forensic Matters

ZL Technologies v. SplitByte (2023)
In this digital forensics, breach of contract, and intellectual property case in the software and data services industry, assisted with forensic analysis of computer systems and artifacts for evidence related to the claims in the pleadings.

Arconic Corp. and Howmet Aerospace Inc. v. Novelis Inc. and Novelis Corp. (2022)
In this trade secret misappropriation and breach of contract case in the aluminum industry, extracted and analyzed the metadata contained in court docket items to demonstrate that the "Author" metadata field does not establish the individual that created the contents of a given Microsoft Word document. Drafted and signed a Declaration to support counsel's successful opposition to a Motion to Recuse.

Deere & Company v. AGCO Corporation (Consulting) (2021)
In this digital forensics and alleged patent infringement case in the agriculture industry, examined numerous forensic images and artifacts to determine the most likely cause of a small number of allegedly confidential documents flowing from one company to the other.

[party names withheld] (2021)
In this digital forensics and trade secret case in the artificial intelligence and technology industry, assisted counsel in acquiring and processing a digital laptop image, scanning for deleted Windows and Linux files, and attempting recovery of deleted files related to alleged trade secret claims. Ongoing engagement.

HeliumCloud v. KWITU (2021)
In this digital forensics, copyright, and breach of contract case in the non-profit industry, assisted counsel in drafting discovery requests and determining which forensic evidence to acquire and preserve. Issued expert report and deposition testimony. Ongoing engagement.

[party names withheld] (2020)
In this digital forensics matter in the video games industry, developed and applied a forensic analysis methodology for investigating a user's usage activity with respect to specific games and the online store for an Xbox One video game console. Ongoing engagement.

Cumberland Forensic (2018)
Analyzed a hard drive for evidence of copying of protected company data to online file shares, and for usage of the Tor browser. Performed an on-site audit to ensure that protective measures put in place were sufficient to guard against future infractions by staff attempting to bypass company policy and security measures.

Edifice Forensic (2017)
Created a forensic image of a laptop. Searched for evidence of drive-wiping tools and recovery of deleted data.

Elalaily Forensic (2017)
Isolated emails sent or received within a certain date range in PST for production.

Thomas Forensic (2017)
Created a forensic image of a laptop and cell phone. Searched for evidence of drive-wiping tools and recovery of deleted email data. Isolated emails sent or received within a certain date range in PST for production.

Welsh Forensic (2017)
Created a forensic image of an android phone and provided text message and MMS analysis for the client in the form of a forensic report.

Emery Federal Credit Union: Forensic Imaging and Analysis (2016)
Imaged a RAID 10 email server and analyzed extracted Exchange server data. Restored data from a proprietary backup format for imaging and analysis; restored and imaged a virtual machine hard drive (VMDK) for inventory and analysis.

State v. [Minor – name withheld] (2016)
Analyzed evidence provided by the State of Maryland to determine if it could be concluded that emails were sent from Defendant to a school administrator. Filed an affidavit to support a motion *in limine* to prevent paper-printout evidence from being used to verify the sender of the email when better evidence was available and email is easily forged. Served as a testifying expert in court, but the case was dismissed in court just prior testimony due to State failing to meet its burden of proof.

State v. Kelvin Sewell (2016)
Created a forensic image from an iPhone 4 and provided text message and MMS analysis for the client in the form of a forensic report.

Elwood Staffing v. Sandler (2016)
Created a forensic image of a laptop computer and searched for evidence of file deletion, as well as searching for evidence that drive-wiping software had been run. Additionally, searched for evidence that company files and data were taken. Provided written forensics report of all findings to the client.

Patriot Metals v. K-fab (2016)
Analyzed windows event logs and IP addresses for evidence of unauthorized remote access to company systems and servers.

ATOS: Forensic Imaging (2016)
Contracted by ATOS to forensically acquire, image, and inventory twelve computers and one USB device. Provided completed acquisitions to ATOS.

Golden v. Gant (2015)
Reviewed three digital audio recording for metadata inconsistencies or other evidence of alteration or tampering.

In re: Vincent L. Abell (2014)
Forensically acquired and imaged a desktop computer hard drive pursuant to a litigation regarding bankruptcy.

Nabijohn v. ITS (NYS Department of Financial Services) (2014)

In this video forensics case performed frame-by-frame analysis of security system footage combined with motion data to conclude whether video files had been altered or footage could conceivably be missing. Drafted expert report.

Pacific Bioscience v. Nutra Luxe MD (2012)

Assisted as a neutral expert in forensic imagining, analysis, and e-discovery regarding emails from a MacBook. Forensically extracted emails from different sources and loaded into a Concordance database.

General Electric Company v. Mitsubishi Heavy Industries, LTD., et al. (2011)

Assisted in developing an electronic discovery application used to review terabytes of backup data and prepare secure reports for counsel without directly viewing confidential data. The application reduced electronic document review costs by orders of magnitude and countered opposing counsel's claims of undue burden to produce relevant documents. Created extensive test data sets designed to simulate the environment of an enterprise system unrolled from tape backups and perform load testing on the application.

General Electric Company, et al. v. Thomas Wilkins (2011)

In this patent infringement case traveled to Kansas City, MO for inspection and inventory of legacy tape collection. Rebuilt legacy tape drive library systems for data recovery and searches for certain keywords related to the patent and defendant.

Declarations, Affidavits, Reports, and Testimony

Arconic v. Novelis – United States District Court for the Western District of Pennsylvania; Case No. 2:17-cv-1434-JFC. Filed November 3, 2017.

Declaration: Declaration signed January 31, 2022 detailing how "Author" and "Created" date metadata work within Microsoft Word documents, what those metadata establish, and demonstrating the variety of "Authors" listed on court filings in the instant matter.

Arkeyo v. Cummins – United States District Court for the Eastern District of Pennsylvania; Case No. 2:16-cv-04720 (ABB). Filed August 29, 2016.

Declaration: Declaration signed April 4, 2017 detailing the importance of the production of the source code repository for expert review.

[party names withheld] (2022 AAA Arbitration).

Declaration: Declaration signed September 19, 2022 in support of counsel motion to compel additional production.

Benchmark Technologies, Inc. v. Yuqiang Tu et al – United States District Court District of Massachusetts; Case No. 1:22-CV-10227-LTS. Filed February 10, 2022.

Expert Report: Expert Report signed on October 7, 2022 containing affirmative opinions evidence of copying in source code.

Cobra Systems, Inc. v. Unger et al. – United States District Court Central District of California; Case No. 16CV00569-ODW-JEM. Filed March 28, 2016.

Declaration: Declaration signed March 20, 2017 detailing the Abstraction Filtration Comparison test performed as well as evidence of copying between software programs.

Covetrus, Inc. and Veterinary Data Services, Inc. v. Actian Corporation – United States District Court for the District of Maine; Case No. 2:21-cv-00097-LEW. Filed April 6, 2021.

Expert Report: Expert Report signed on June 24, 2022 containing affirmative opinions on comparisons of source code and copyright registrations, responses to Plaintiff's Expert opinions, as well as detailing audit findings.

Testimony: Provided deposition testimony on July 1, 2022 on topics covered in Expert report related to code comparisons, copyright registration, and the systems audit.

ECIMOS, LLC v. Carrier Corporation - United States District Court Western District of Tennessee, Western Division; Case No. 2:15-cv-2776-JPM-cgc. Filed November 6, 2015.

Declaration: Declaration signed December 22, 2016 in rebuttal to Plaintiff's claims, as well as detailing the challenges of comparing text-based source code to graphical source code.

Testimony: Testimony provided at a September 1, 2017 Preliminary Injunction Hearing reiterating points made in the Declaration, as well as touching on elements of proper clean room design and details of the source code review.

Declaration: Declaration signed October 13, 2017 in support of Carrier's memorandum in opposition to Plaintiff's motion to reopen proof, detailing material issues with the opposing expert's report.

Expert Report: Expert Report signed on October 30, 2017 containing affirmative opinions as well as detailing material issues with the opposing expert's report.

Testimony: Provided deposition testimony on January 11, 2018 on topics covered in Declarations and Expert report related to clean room design, database comparison and source code analysis.

Supplemental Expert Report: Expert Report signed on April 30, 2018 containing affirmative opinions as well as rebutting the newest assertions from opposing expert. Abstraction Filtration Comparison test included with regard to the databases at issue in this case.

Testimony: Testimony provided at June 29, 2018 jury trial reiterating points made in the Declarations as well as Expert reports and exhibits. Specific focus on database and software issues regarding copyrights and trade secrets, and the Abstraction Filtration Comparison test.

HeliumCloud v. KWITU – United States District Court for the District of Maryland; Case No. 8:2021cv01212. Filed May 17, 2021.

Expert Report: Expert Report signed March 14, 2022 detailing deficiencies in the production and preservation of evidence, as well as the analyses that would be performed on the evidence, had it been properly preserved and produced in the instant matter.

Testimony: Provided deposition testimony on October 20, 2022 on topics covered in Expert report related to deficiencies in the production and preservation of evidence.

Nexus v. Krougly, Limesoft et al. – Ontario Superior Court of Justice; Court File No. 3660/2011. Filed March 8, 2011.

Expert Report: Expert Report signed April, 6, 2022 detailing how source code repositories were connected to and used for software development on from two computers based on forensic evidence on the two computers.

Affidavit: Affidavit signed February 15, 2017 detailing how to extract and restore data from an SVN repository, as well as the importance of specific xml files in validating the origin of produced source code.

Testimony: Testimony provided at a hearing on July 13, 2017 covering the topics explained in the Affidavit, as well as touching briefly on clean room design.

Affidavit: Affidavit signed December 16, 2021 detailing the presence of forensic artifacts related to e-mail migration from Microsoft Outlook to Google Apps, as well as detailing forensic artifacts referencing to specific e-mail addresses of interest.

Trent P. Fisher Enters. v. SAS Automation, LLC – United States District Court Southern District of Ohio; Case No. 3:20-cv-216. Filed March 31, 2021.

Expert Report: Expert Report signed November 11, 2022 detailing how the GPL should apply to central issues in the matter, as well as responding to Plaintiff's expert report and opinions contained therein.

Testimony: Provided deposition testimony on January 31, 2023 on topics covered in Expert report related to responses to opposing expert's opinions, the GNU General Public License, version 3, and the software at issue in this case.

TruLogic, Inc. v. General Electric Company through its GE Aviation Division – Common Pleas Court of Greene County, Ohio General Division; Case No. 2020 CV 0464. Filed September 15, 2020.

Expert Report: Expert Report signed June 12, 2023 addressing quantity and quality of alleged copying, as well as responding to Plaintiff's expert report and opinions contained therein.

Supplemental Expert Report: Supplemental Expert Report signed July 20, 2023 addressing factual findings, screenshots, as well as addressing Plaintiff's expert second supplemental report.

State v. [Minor – name withheld] – Montgomery County, MD Circuit Court; Petition #06-J-16-050314. Filed October, 2016.

Affidavit: Affidavit signed November 14, 2016 detailing the steps one could take in order to authenticate that an email was sent from a specific device and received by the recipient, and that paper printouts, in lieu of any other qualifying information, were not sufficient to authenticate an email.

EDUCATION AND EMPLOYMENT HISTORY

Education

Bachelor of Arts, Computer Science
Certificate in Bioinformatics and Modeling
The Wesleyan University, Middletown, CT

Employment History

Testifying Expert, Manager and Forensic Examiner, DisputeSoft (Jan. 2011–Present)
Drafted Affidavits, Declarations, and Expert Reports to support counsel. Testified as an expert in areas related to software, copyright, and computer systems. Performed and is knowledgeable in the Abstraction Filtration Comparison test. Forensically acquired, rebuilt, and tested numerous system environments. Performed various analyses on data extracted from a variety of database types and systems. Drafted expert reports and aided in formulating expert witness opinions for cases. Performed root cause analyses related to system outages pursuant to SLAs and MSA requirements. Built and administered a domain from the ground up, including Active Directory, Group policy, VPN, telephony, wired and wireless networking, Disaster Recovery and backup strategy, antivirus and SharePoint solutions. Managed updates, backups and recovery for all server data and systems, as well as system security for an office of networked and computers. Managed all hardware capacity planning, implementation, maintenance and support, as well as supporting and maintaining software licenses and warranties. Certified as an EnCase Certified Examiner for computer forensics since October, 2012.

IT Manager and Systems Administrator, TM Associates Management, Inc (Oct. 2005-Jan. 2011)

Managed updates, backups, and recovery for all server data and systems. Maintained system security, Active Directory and domain services for an office network of computers as well as for over 150 remote locations, including protection against and removal of viruses and malware. Maintained critical system application servers in OS/400 as well as Windows environments. Provided system and application support to over 100 users for a custom-based property management software as well as Windows operating systems. Created and customized a number of Crystal Reports and administered the database for the proprietary software solution. Managed all hardware capacity planning, implementation, maintenance and support, as well as supporting and maintaining software licenses and warranties.

Certifications

EnCase Certified Examiner (since October 2012)

Programming Language and Database Familiarity

Java, C#, C, CPP, Objective-C/Swift, Python
MS-SQL, MySQL/MariaDB, Oracle, SQLite

ATTACHMENT 2

LIST OF DOCUMENTS CONSIDERED

BOSCH000082
BOSCH000105
BOSCH000923
BOSCH000924
BOSCH001362
BOSCH002654
BOSCH002656
BOSCH002764
BOSCH002770
BOSCH002773
BOSCH002775
BOSCH002778
BOSCH002795
BOSCH002827
CS00444978
CS00013052
CS00437680
CS00016681

ATTACHMENT 3
RELEASE DATES FOR BOSCH CDR SOFTWARE VERSIONS

Bosch CDR Software Version	Date of Release
v23.1	03/31/2023
v23.0	10/7/2022
v21.5	4/8/2022
v21.4	1/5/2022
v21.3	10/28/2021
v21.2	8/3/2021
v21.1	5/14/2021
v21.0	1/21/2021
v19.6	12/1/2020
v19.5	8/13/2020
v19.4	5/7/2020
v19.3	12/23/2019
v19.2	12/6/2019
v19.1	9/20/2019
v19.0	6/6/2019
v18.0	2/20/2019
v17.10	12/20/2018
v17.9	9/18/2018
v17.8	7/31/2018
v17.7	3/19/2018
v17.6	12/22/2017
v17.5	10/17/2017
v17.4	6/15/2017
v17.3	4/11/2017
v17.2	1/22/2017
v17.1	11/2/2016
v17.0	8/23/2016
v16.6	5/23/2016
v16.5	3/18/2016
v16.4	12/17/2015
v16.3	11/20/2015
v16.2	8/10/2015

DisputeSoft
Software Dispute Experts

6116 Executive Blvd., Suite 330, North Bethesda, MD 20852
main **301.251.6313** · cell **518.982.7344** · fax **240.465.4442**
rsubbu@disputesoft.com
disputesoft.com

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

Invoice #: 22AU-413T
Date: 09/08/2022
Invoice Due Date: 10/23/2022
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For Professional Services Rendered:

Period: August 1, 2022 to August 31, 2022

Total Fees	$17,472.00
Total Expenses	$421.74
New Charges:	$17,893.74
Outstanding Balance:	$0.00
New Balance:	**$17,893.74**

Payment by ACH/Wire:

Beneficiary Name:	Taron, LLC dba DisputeSoft
Bank SWIFT - Name:	KEYBUS33 - KeyBank N. A.
Bank Routing:	021300077
Account Number:	776871001553

Account Summary						
Services BTD	Expenses BTD	Last Inv Num	Last Inv Date	Last Inv Amt	Last Pay Amt	Prev Unpaid Amt
$25,716.00	$632.68	22JL-413T	8/4/2022	$3,744.00	$117.00	$0.00

Total Amount Due Including This Invoice: **$17,893.74**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

The following is a summary of time and expenses for professional services rendered on this
Matter for the period: August 1, 2022 to August 31, 2022

Fees by Staff:

Name	Hours	Rate	Amount
Joshua A. Siegel	44.8	$390.00	$17,472.00
Total Hours and Fees	**44.8**		**$17,472.00**

Expenses by Staff:

Name	Amount
Joshua A. Siegel	$421.74
Total Out-of-Pocket Expenses	**$421.74**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

For professional services rendered for the period August 1, 2022 to August 31, 2022

Date	Staff	Description	Hrs
08/02/2022	JAS	Draft TeamViewer setup directions	1.1
08/03/2022	JAS	Research viable teamviewer alternatives and potential problems	0.8
08/09/2022	JAS	Setup and test screen recording software, configure review computer, encrypt and test external drive	1.3
08/10/2022	JAS	Setup and test screen recording software, configure review computer, encrypt and test external drive	1.5
08/11/2022	JAS	Setup and test screen recording software, configure review computer, encrypt and test external drive	0.5
08/15/2022	JAS	Setup and test screen recording software, configure review computer, encrypt and test external drive	2.8
08/15/2022	JAS	Connect to audit computer, begin indexing	0.3
08/16/2022	JAS	Setup and test screen recording software, configure review computer, encrypt and test external drive	0.7
08/16/2022	JAS	Audit of computer	4.9
08/17/2022	JAS	Review, update, and consolidate audit notes	0.4
08/17/2022	JAS	Setup and test screen recording software, configure review computer, encrypt and test external drive	0.5
08/17/2022	JAS	Audit of computer	3.6
08/18/2022	JAS	Review, update, and consolidate audit notes	0.6
08/18/2022	JAS	Setup and test screen recording software, configure review computer, encrypt and test external drive	0.5
08/18/2022	JAS	Audit of computer	4.5
08/19/2022	JAS	Audit of computer	4.5
08/19/2022	JAS	Review, update, and consolidate audit notes	2.1
08/19/2022	JAS	Setup and test screen recording software, configure review computer, encrypt and test external drive	0.3
08/22/2022	JAS	Call with counsel for status updates	1.5
08/22/2022	JAS	Review, update, and consolidate audit notes	4.1
08/24/2022	JAS	Audit of computer	0.3
08/24/2022	JAS	Review, update, and consolidate audit notes	1.5
08/25/2022	JAS	Audit of computer	3.7
08/25/2022	JAS	Review, update, and consolidate audit notes	0.8
08/26/2022	JAS	Audit of computer	0.8
08/26/2022	JAS	Review, update, and consolidate audit notes	0.6
08/29/2022	JAS	Review, update, and consolidate audit notes	0.6

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

	Total Hours for Period	**44.8**

Expenses for the period: August 1, 2022 to August 31, 2022

Date	Staff	Description	Amount
08/15/2022	JAS	Team Viewer	$421.74
		Total Expenses for Period	**$421.74**

TeamViewer

Hi Josh Heim Siegel

Thank you for choosing TeamViewer. We have received your order 77UN00050702, summarized below.

What happens next?

After your order has been processed, you will receive separate emails with your license activation link, invoice, and Customer Portal account – where you can manage your TeamViewer subscription(s) online.

Order Summary

Contact Information

Josh Helm Siegel

jsiegel@disputesoft.com

Billing Address

DisputeSoft

6116 Executive

Boulevard, Suite 330

20852-4935 North

Bethesda

Maryland

United States

Subscription*

Order
Date: 15 August
           2022
Billing   15 August
Period: 2022 to
           14 August
           2023

Payment Method

Credit Card

Description	QTY	Amount
TeamViewer Business	1	~~418.80~~
• 1 licensed user that can start connections from 3 designated devices		397.86
• Access to unlimited devices		
• 1 concurrent connection (channel)		
• Up to 3 concurrent sessions per channel in separate tabs		
• 200 managed devices		
• 1 organizer that can host a meeting with 10 participants		

Subtotal (USD) 397.86

8% Tax (USD)   23.88

Total (USD) 421.74

*Your subscription will automatically renew every 12 months,
unless you terminate your contract by submitting a support ticket
or notifying us in written or text form at least 28 days before the

DisputeSoft
Software Dispute Experts

6116 Executive Blvd., Suite 330, North Bethesda, MD 20852
main **301.251.6313** · cell **518.982.7344** · fax **240.465.4442**
rsubbu@disputesoft.com
disputesoft.com

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

Invoice #: 22SE-413T
Date: 10/07/2022
Invoice Due Date: 11/21/2022
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For Professional Services Rendered:

Period: September 1, 2022 to September 30, 2022

Total Fees	$3,549.00
Total Expenses	$0.00
New Charges:	$3,549.00
Outstanding Balance:	$17,893.74
New Balance:	**$21,442.74**

Payment by ACH/Wire:

Beneficiary Name:	Taron, LLC dba DisputeSoft
Bank SWIFT - Name:	KEYBUS33 - KeyBank N. A.
Bank Routing:	021300077
Account Number:	776871001553

Account Summary						
Services BTD	Expenses BTD	Last Inv Num	Last Inv Date	Last Inv Amt	Last Pay Amt	Prev Unpaid Amt
$29,265.00	$632.68	22AU-413T	9/8/2022	$17,893.74	$117.00	$17,893.74

Total Amount Due Including This Invoice: **$21,442.74**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

The following is a summary of time and expenses for professional services rendered on this
Matter for the period: September 1, 2022 to September 30, 2022

Fees by Staff:

Name	Hours	Rate	Amount
Joshua A. Siegel	9.1	$390.00	$3,549.00
Total Hours and Fees	**9.1**		**$3,549.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

For professional services rendered for the period September 1, 2022 to September 30, 2022

Date	Staff	Description	Hrs
09/12/2022	JAS	Call with counsel re: Audit	0.3
09/16/2022	JAS	Draft and revise declaration	6.0
09/19/2022	JAS	Draft and revise declaration	2.8
		Total Hours for Period	9.1

DisputeSoft
Software Dispute Experts

6116 Executive Blvd., Suite 330, North Bethesda, MD 20852
main **301.251.6313** · cell **518.982.7344** · fax **240.465.4442**
rsubbu@disputesoft.com
disputesoft.com

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

*Invoice #:*22OC-413T
Date: 11/07/2022
Invoice Due Date: 12/22/2022
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For Professional Services Rendered:

Period: October 1, 2022 to October 31, 2022

Total Fees	$585.00
Total Expenses	($23.88)
New Charges:	$561.12
Outstanding Balance:	$0.00
New Balance:	**$561.12**

Payment by ACH/Wire:

Beneficiary Name:	Taron, LLC dba DisputeSoft
Bank SWIFT - Name:	KEYBUS33 - KeyBank N. A.
Bank Routing:	021300077
Account Number:	776871001553

Account Summary						
Services BTD	Expenses BTD	Last Inv Num	Last Inv Date	Last Inv Amt	Last Pay Amt	Prev Unpaid Amt
$29,850.00	$608.80	22SE-413T	10/7/2022	$3,549.00	$3,549.00	$0.00

Total Amount Due Including This Invoice: **$561.12**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

*Invoice #:*22OC-413T
Date: 11/07/2022
Invoice Due Date: 12/22/2022
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

The following is a summary of time and expenses for professional services rendered on this
Matter for the period: October 1, 2022 to October 31, 2022

Fees by Staff:

Name	Hours	Rate	Amount
Joshua A. Siegel	1.5	$390.00	$585.00
Total Hours and Fees	**1.5**		**$585.00**

Expenses by Staff:

Name	Amount
Raj Subbu	($23.88)
Total Out-of-Pocket Expenses	**($23.88)**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

Invoice #: 22OC-413T
Date: 11/07/2022
Invoice Due Date: 12/22/2022
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For professional services rendered for the period October 1, 2022 to October 31, 2022

Date	Staff	Description	Hrs
10/11/2022	JAS	Setup and test new BitBucket access	0.4
10/14/2022	JAS	Answer counsel questions	0.5
10/21/2022	JAS	Assist counsel with search suggestion terms and areas	0.6
		Total Hours for Period	**1.5**

Expenses for the period: October 1, 2022 to October 31, 2022

Date	Staff	Description	Amount
10/12/2022	RS	Refund of MD Sales Tax on Commercial Use of TeamViewer Purchased 8/15/22 - MD Senate Bill 723	($23.88)
		Total Expenses for Period	**($23.88)**

DisputeSoft
Software Dispute Experts

6116 Executive Blvd., Suite 330, North Bethesda, MD 20852
main **301.251.6313** · cell **518.982.7344** · fax **240.465.4442**
rsubbu@disputesoft.com
disputesoft.com

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

Invoice #: 22NV-413T
Date: 12/07/2022
Invoice Due Date: 01/21/2023
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For Professional Services Rendered:

Period: November 1, 2022 to November 30, 2022

Total Fees	$2,028.00
Total Expenses	$0.00
New Charges:	$2,028.00
Outstanding Balance:	$561.12
New Balance:	**$2,589.12**

Payment by ACH/Wire:

Beneficiary Name:	Taron, LLC dba DisputeSoft
Bank SWIFT - Name:	KEYBUS33 - KeyBank N. A.
Bank Routing:	021300077
Account Number:	776871001553

Account Summary						
Services BTD	Expenses BTD	Last Inv Num	Last Inv Date	Last Inv Amt	Last Pay Amt	Prev Unpaid Amt
$31,878.00	$608.80	22OC-413T	11/7/2022	$561.12	$3,549.00	$561.12

	Total Amount Due Including This Invoice:	**$2,589.12**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

The following is a summary of time and expenses for professional services rendered on this
Matter for the period: November 1, 2022 to November 30, 2022

Fees by Staff:

Name	Hours	Rate	Amount
Joshua A. Siegel	5.2	$390.00	$2,028.00
Total Hours and Fees	**5.2**		**$2,028.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

For professional services rendered for the period November 1, 2022 to November 30, 2022

Date	Staff	Description	Hrs
11/28/2022	JAS	Prep system for bitbucket audit, setup initial document, review results and summarize info	1.5
11/28/2022	JAS	Bitbucket Audit	1.1
11/29/2022	JAS	Prep system for bitbucket audit, setup initial document, review results and summarize info	1.1
11/29/2022	JAS	Bitbucket Audit	0.8
11/30/2022	JAS	Prep system for bitbucket audit, setup initial document, review results and summarize info	0.7
		Total Hours for Period	**5.2**

DisputeSoft
Software Dispute Experts

6116 Executive Blvd., Suite 330, North Bethesda, MD 20852
main **301.251.6313** · cell **518.982.7344** · fax **240.465.4442**
rsubbu@disputesoft.com
disputesoft.com

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

Invoice #: 22DE-413T
Date: 01/06/2023
Invoice Due Date: 02/20/2023
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For Professional Services Rendered:

Period: December 1, 2022 to December 31, 2022

Total Fees	$10,890.00
Total Expenses	$0.00
New Charges:	$10,890.00
Outstanding Balance:	$0.00
New Balance:	**$10,890.00**

Payment by ACH/Wire:

Beneficiary Name:	Taron, LLC dba DisputeSoft
Bank SWIFT - Name:	KEYBUS33 - KeyBank N. A.
Bank Routing:	021300077
Account Number:	776871001553

Account Summary						
Services BTD	Expenses BTD	Last Inv Num	Last Inv Date	Last Inv Amt	Last Pay Amt	Prev Unpaid Amt
$42,768.00	$608.80	22NV-413T	12/7/2022	$2,028.00	$561.12	$0.00

Total Amount Due Including This Invoice: **$10,890.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

The following is a summary of time and expenses for professional services rendered on this
Matter for the period: December 1, 2022 to December 31, 2022

Fees by Staff:

Name	Hours	Rate	Amount
Joshua A. Siegel	27.5	$390.00	$10,725.00
Thomas J. Wolf	0.6	$275.00	$165.00
Total Hours and Fees	**28.1**		**$10,890.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

For professional services rendered for the period December 1, 2022 to December 31, 2022

Date	Staff	Description	Hrs
12/01/2022	JAS	Prep system for bitbucket audit, setup initial document, review results and summarize info	0.5
12/01/2022	JAS	Bitbucket Audit	0.9
12/02/2022	JAS	Prep system for bitbucket audit, setup initial document, review results and summarize info	0.7
12/02/2022	JAS	Bitbucket Audit	2.3
12/05/2022	JAS	Prep system for bitbucket audit, setup initial document, review results and summarize info	0.5
12/06/2022	JAS	Prep system for bitbucket audit, setup initial document, review results and summarize info	0.3
12/06/2022	JAS	Call with counsel	1.1
12/07/2022	JAS	Setup initial document for AWS audit, review results and summarize info	0.5
12/07/2022	TJW	Review and sign protective order	0.6
12/07/2022	JAS	Amazon Web Services and RDS Audit	1.4
12/08/2022	JAS	Researching AWS Elastic Beanstalk, VPC, EC2, and RDS options and connectivity	1.8
12/08/2022	JAS	Amazon Web Services and RDS Audit	1.8
12/08/2022	JAS	Troubleshooting AWS database connection	1.1
12/09/2022	JAS	Troubleshooting AWS database connection	0.7
12/09/2022	JAS	Researching AWS Elastic Beanstalk, VPC, EC2, and RDS options and connectivity	0.6
12/09/2022	JAS	Review and research audit data in re: replay source	1.1
12/12/2022	JAS	Amazon Web Services and RDS Audit	0.7
12/12/2022	JAS	Troubleshooting AWS database connection	0.3
12/12/2022	JAS	Review and research audit data in re: replay source	1.4
12/14/2022	JAS	Review and research audit data in re: replay source	1.0
12/15/2022	JAS	Review and research audit data in re: replay source	0.4
12/21/2022	JAS	Source code repository audit: CDR data	0.8
12/21/2022	JAS	Review and research audit data in re: replay source	0.7
12/22/2022	JAS	Review and research audit data in re: replay source	0.8
12/22/2022	JAS	Source code repository audit: CDR data	1.5
12/27/2022	JAS	Source code repository audit: CDR data	0.3
12/28/2022	JAS	Drafting Bosch Audit Report	3.3

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

Invoice #: 22DE-413T
Date: 01/06/2023
Invoice Due Date: 02/20/2023
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

12/29/2022	JAS	Drafting Bosch Audit Report	1.0
		Total Hours for Period	**28.1**

DisputeSoft
Software Dispute Experts

6116 Executive Blvd., Suite 330, North Bethesda, MD 20852
main **301.251.6313** · cell **518.982.7344** · fax **240.465.4442**
rsubbu@disputesoft.com
disputesoft.com

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

*Invoice #:*23JA-413T
Date: 02/06/2023
Invoice Due Date: 03/23/2023
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For Professional Services Rendered:

Period: January 1, 2023 to January 31, 2023

Total Fees	$8,853.00
Total Expenses	$0.00
New Charges:	$8,853.00
Outstanding Balance:	$0.00
New Balance:	**$8,853.00**

Payment by ACH/Wire:	
Beneficiary Name:	Taron, LLC dba DisputeSoft
Bank SWIFT - Name:	KEYBUS33 - KeyBank N. A.
Bank Routing:	021300077
Account Number:	776871001553

Account Summary						
Services BTD	Expenses BTD	Last Inv Num	Last Inv Date	Last Inv Amt	Last Pay Amt	Prev Unpaid Amt
$51,621.00	$608.80	22DE-413T	1/6/2023	$10,890.00	$10,890.00	$0.00

Total Amount Due Including This Invoice:	**$8,853.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

The following is a summary of time and expenses for professional services rendered on this
Matter for the period: January 1, 2023 to January 31, 2023

Fees by Staff:

Name	Hours	Rate	Amount
Joshua A. Siegel	22.7	$390.00	$8,853.00
Total Hours and Fees	**22.7**		**$8,853.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

For professional services rendered for the period January 1, 2023 to January 31, 2023

Date	Staff	Description	Hrs
01/04/2023	JAS	Draft Audit Findings Report	1.3
01/05/2023	JAS	Draft Audit Findings Report	0.7
01/06/2023	JAS	Draft Audit Findings Report	1.2
01/09/2023	JAS	Draft Audit Findings Report	2.2
01/13/2023	JAS	Draft Audit Findings Report	3.2
01/20/2023	JAS	Draft audit findings report	2.5
01/23/2023	JAS	Call with Counsel	0.9
01/23/2023	JAS	Review extension listing for new production prior to call with counsel	0.5
01/23/2023	JAS	Draft Audit Findings Report	0.8
01/24/2023	JAS	Draft Audit Findings Report	1.0
01/25/2023	JAS	Review extension listing for new production prior to call with counsel	1.1
01/26/2023	JAS	Review extension listing for new production prior to call with counsel	0.6
01/26/2023	JAS	Draft Audit Findings Report	1.4
01/26/2023	JAS	Review new EULAs and terms	0.7
01/27/2023	JAS	Draft Audit Findings Report	4.6
		Total Hours for Period	**22.7**

DisputeSoft
Software Dispute Experts

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

Invoice #: 23FE-413T
Date: 03/08/2023
Invoice Due Date: 04/22/2023
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For Professional Services Rendered:

Period: February 1, 2023 to February 28, 2023

Total Fees	$1,811.00
Total Expenses	$0.00
New Charges:	$1,811.00
Outstanding Balance:	$0.00
New Balance:	**$1,811.00**

Payment by ACH/Wire:	
Beneficiary Name:	Taron, LLC dba DisputeSoft
Bank SWIFT - Name:	KEYBUS33 - KeyBank N. A.
Bank Routing:	021300077
Account Number:	776871001553

Account Summary						
Services BTD	Expenses BTD	Last Inv Num	Last Inv Date	Last Inv Amt	Last Pay Amt	Prev Unpaid Amt
$53,432.00	$608.80	23JA-413T	2/6/2023	$8,853.00	$8,853.00	$0.00

Total Amount Due Including This Invoice: **$1,811.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

<div align="right">

*Invoice #:*23FE-413T
Date: 03/08/2023
Invoice Due Date: 04/22/2023
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

</div>

The following is a summary of time and expenses for professional services rendered on this
Matter for the period: February 1, 2023 to February 28, 2023

Fees by Staff:

Name	Hours	Rate	Amount
Joshua A. Siegel	1.4	$390.00	$546.00
Thomas J. Wolf	4.6	$275.00	$1,265.00
Total Hours and Fees	**6.0**		**$1,811.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

For professional services rendered for the period February 1, 2023 to February 28, 2023

Date	Staff	Description	Hrs
02/10/2023	JAS	Review new production re: Reverse Engineering	1.1
02/14/2023	JAS	Review new production documents	0.3
02/15/2023	TJW	Review and revise draft of J.Siegel audit report	1.8
02/15/2023	TJW	Review and revise draft of audit report	1.6
02/16/2023	TJW	Review and revise draft of J.Siegel audit report	0.8
02/17/2023	TJW	Review and revise draft of J.Siegel audit report	0.4
		Total Hours for Period	6.0

DisputeSoft
Software Dispute Experts

6116 Executive Blvd., Suite 330, North Bethesda, MD 20852
main **301.251.6313** · cell **518.982.7344** · fax **240.465.4442**
rsubbu@disputesoft.com
disputesoft.com

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

Invoice #: 23MR-413T
Date: 04/07/2023
Invoice Due Date: 05/22/2023
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For Professional Services Rendered:

Period: March 1, 2023 to March 31, 2023

Total Fees	$2,457.00
Total Expenses	$0.00
New Charges:	$2,457.00
Outstanding Balance:	$0.00
New Balance:	**$2,457.00**

Payment by ACH/Wire:

Beneficiary Name:	Taron, LLC dba DisputeSoft
Bank SWIFT - Name:	KEYBUS33 - KeyBank N. A.
Bank Routing:	021300077
Account Number:	776871001553

Account Summary						
Services BTD	Expenses BTD	Last Inv Num	Last Inv Date	Last Inv Amt	Last Pay Amt	Prev Unpaid Amt
$55,889.00	$608.80	23FE-413T	3/8/2023	$1,811.00	$1,811.00	$0.00

Total Amount Due Including This Invoice: **$2,457.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

The following is a summary of time and expenses for professional services rendered on this
Matter for the period: March 1, 2023 to March 31, 2023

Fees by Staff:

Name	Hours	Rate	Amount
Joshua A. Siegel	6.3	$390.00	$2,457.00
Total Hours and Fees	**6.3**		**$2,457.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

For professional services rendered for the period March 1, 2023 to March 31, 2023

Date	Staff	Description	Hrs
03/21/2023	JAS	Call with counsel to review Audit findings report	0.7
03/21/2023	JAS	Review recent documents to prepare for Bosch Call with Counsel	0.8
03/22/2023	JAS	Drafting Audit report	0.5
03/30/2023	JAS	Draft audit findings report	3.8
03/31/2023	JAS	Draft audit findings report	0.5
		Total Hours for Period	**6.3**

DisputeSoft
Software Dispute Experts

6116 Executive Blvd., Suite 330, North Bethesda, MD 20852
main **301.251.6313** · cell **518.982.7344** · fax **240.465.4442**
rsubbu@disputesoft.com
disputesoft.com

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington Hills, MI 48331

Invoice #: 23AP-413T
Date: 05/08/2023
Invoice Due Date: 06/22/2023
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For Professional Services Rendered:

Period: April 1, 2023 to April 30, 2023

Total Fees	$11,895.00
Total Expenses	$0.00
New Charges:	$11,895.00
Outstanding Balance:	$0.00
New Balance:	**$11,895.00**

Payment by ACH/Wire:

Beneficiary Name:	Taron, LLC dba DisputeSoft
Bank SWIFT - Name:	KEYBUS33 - KeyBank N. A.
Bank Routing:	021300077
Account Number:	776871001553

Account Summary						
Services BTD	Expenses BTD	Last Inv Num	Last Inv Date	Last Inv Amt	Last Pay Amt	Prev Unpaid Amt
$67,784.00	$608.80	23MR-413T	4/7/2023	$2,457.00	$2,457.00	$0.00

Total Amount Due Including This Invoice: **$11,895.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington Hills, MI 48331

The following is a summary of time and expenses for professional services rendered on this
Matter for the period: April 1, 2023 to April 30, 2023

Fees by Staff:

Name	Hours	Rate	Amount
Joshua A. Siegel	30.5	$390.00	$11,895.00
Total Hours and Fees	**30.5**		**$11,895.00**

For professional services rendered for the period April 1, 2023 to April 30, 2023

Date	Staff	Description	Hrs
04/03/2023	JAS	Draft Audit Findings Report	0.6
04/04/2023	JAS	Draft Audit Findings Report	2.8
04/06/2023	JAS	Draft Audit Findings Report	0.5
04/07/2023	JAS	Draft Audit Findings Report	3.4
04/11/2023	JAS	Draft Bosch Audit report	2.3
04/12/2023	JAS	Draft Bosch Audit report	0.7
04/13/2023	JAS	Draft Bosch Audit report	2.1
04/14/2023	JAS	Draft Bosch Audit report	3.4
04/17/2023	JAS	Draft Bosch Audit report	2.5
04/18/2023	JAS	Draft Bosch Audit report	1.1
04/19/2023	JAS	Draft Bosch Audit report	2.7
04/20/2023	JAS	Draft Bosch Audit report	1.2
04/21/2023	JAS	Draft Bosch Audit report	3.2
04/27/2023	JAS	Draft and update Audit Report	1.3
04/28/2023	JAS	Draft and update Audit Report	2.7

Total Hours for Period   **30.5**

DisputeSoft
Software Dispute Experts

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington Hills, MI 48331

Invoice #: 23MY-413T
Date: 06/08/2023
Invoice Due Date: 07/23/2023
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For Professional Services Rendered:

Period: May 1, 2023 to May 31, 2023

Total Fees	$21,507.00
Total Expenses	$0.00
New Charges:	$21,507.00
Outstanding Balance:	$0.00
New Balance:	**$21,507.00**

Payment by ACH/Wire:	
Beneficiary Name:	Taron, LLC dba DisputeSoft
Bank SWIFT - Name:	KEYBUS33 - KeyBank N. A.
Bank Routing:	021300077
Account Number:	776871001553

Account Summary						
Services BTD	Expenses BTD	Last Inv Num	Last Inv Date	Last Inv Amt	Last Pay Amt	Prev Unpaid Amt
$89,291.00	$608.80	23AP-413T	5/8/2023	$11,895.00	$11,895.00	$0.00

Total Amount Due Including This Invoice:	**$21,507.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington Hills, MI 48331

The following is a summary of time and expenses for professional services rendered on this
Matter for the period: May 1, 2023 to May 31, 2023

Fees by Staff:

Name	Hours	Rate	Amount
Joshua A. Siegel	49.3	$390.00	$19,227.00
Anne D. Ackerman	4.3	$300.00	$1,290.00
Thomas J. Wolf	3.6	$275.00	$990.00
Total Hours and Fees	**57.2**		**$21,507.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington Hills, MI 48331

*Invoice #:*23MY-413T
Date: 06/08/2023
Invoice Due Date: 07/23/2023
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For professional services rendered for the period May 1, 2023 to May 31, 2023

Date	Staff	Description	Hrs
05/01/2023	JAS	Draft and update Audit Report	2.2
05/02/2023	JAS	Draft and update Audit Report	1.7
05/03/2023	JAS	Draft and update Audit Report	1.3
05/04/2023	JAS	Draft and update Audit Report	3.3
05/05/2023	JAS	Draft and update Audit Report	3.7
05/08/2023	JAS	Draft and update Audit Report	2.6
05/09/2023	JAS	Draft and update Audit Report	4.1
05/10/2023	JAS	Draft and update Audit Report	5.1
05/10/2023	TJW	Assist with review and revising draft of audit report	1.7
05/10/2023	TJW	Review and revise draft of J.Siegel software audit report	1.9
05/11/2023	JAS	Draft and update Audit Report	2.3
05/11/2023	ADA	Assist with review and edit of draft expert report	2.7
05/11/2023	ADA	Assist with review of citations of draft expert report	1.6
05/12/2023	JAS	Draft and update Audit Report	3.1
05/15/2023	JAS	Draft and update Audit Report	2.1
05/16/2023	JAS	Draft and update Audit Report	1.0
05/22/2023	JAS	Review new production	0.5
05/22/2023	JAS	Draft Audit report	0.5
05/23/2023	JAS	Call with counsel for status update	1.7
05/23/2023	JAS	Draft Audit report	0.8
05/23/2023	JAS	Review new production	1.1
05/24/2023	JAS	Draft Audit report	2.3
05/25/2023	JAS	Draft Audit report	2.7
05/26/2023	JAS	Validate software version release dates	0.4
05/26/2023	JAS	Draft Audit report	3.3
05/26/2023	JAS	Create and format report Attachments	2.1
05/26/2023	JAS	Review new production	1.4
		Total Hours for Period	**57.2**

DisputeSoft
Software Dispute Experts

6116 Executive Blvd., Suite 330, North Bethesda, MD 20852
main **301.251.6313** · cell **518.982.7344** · fax **240.465.4442**
rsubbu@disputesoft.com
disputesoft.com

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington Hills, MI 48331

Invoice #: 23JN-413T
Date: 07/08/2023
Invoice Due Date: 08/22/2023
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For Professional Services Rendered:

Period: June 1, 2023 to June 30, 2023

Total Fees	$1,950.00
Total Expenses	$0.00
New Charges:	$1,950.00
Outstanding Balance:	$0.00
New Balance:	**$1,950.00**

Payment by ACH/Wire:

Beneficiary Name:	Taron, LLC dba DisputeSoft
Bank SWIFT - Name:	KEYBUS33 - KeyBank N. A.
Bank Routing:	021300077
Account Number:	776871001553

Account Summary						
Services BTD	Expenses BTD	Last Inv Num	Last Inv Date	Last Inv Amt	Last Pay Amt	Prev Unpaid Amt
$91,241.00	$608.80	23MY-413T	6/8/2023	$21,507.00	$21,507.00	$0.00

Total Amount Due Including This Invoice: **$1,950.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington Hills, MI 48331

*Invoice #:*23JN-413T
Date: 07/08/2023
Invoice Due Date: 08/22/2023
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

The following is a summary of time and expenses for professional services rendered on this
Matter for the period: June 1, 2023 to June 30, 2023

Fees by Staff:

Name	Hours	Rate	Amount
Joshua A. Siegel	5.0	$390.00	$1,950.00
Total Hours and Fees	**5.0**		**$1,950.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington Hills, MI 48331

For professional services rendered for the period June 1, 2023 to June 30, 2023

Date	Staff	Description	Hrs
06/19/2023	JAS	Review, revise and edit Audit report	1.1
06/19/2023	JAS	Encrypted email research and setup per Leone Apollonia	1.7
06/20/2023	JAS	Review, revise and edit Audit report	0.3
06/26/2023	JAS	Revise and update Audit report	0.8
06/26/2023	JAS	Case status updates for counsel	0.6
06/27/2023	JAS	Revise and update Audit report	0.5
		Total Hours for Period	**5.0**

DisputeSoft
Software Dispute Experts

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington Hills, MI 48331

Invoice #: 23JL-413T
Date: 08/04/2023
Invoice Due Date: 09/18/2023
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For Professional Services Rendered:

Period: July 1, 2023 to July 31, 2023

Total Fees	$19,652.00
Total Expenses	$0.00
New Charges:	$19,652.00
Outstanding Balance:	$1,950.00
New Balance:	**$21,602.00**

Payment by ACH/Wire:

Beneficiary Name:	Taron, LLC dba DisputeSoft
Bank SWIFT - Name:	KEYBUS33 - KeyBank N. A.
Bank Routing:	021300077
Account Number:	776871001553

Account Summary						
Services BTD	Expenses BTD	Last Inv Num	Last Inv Date	Last Inv Amt	Last Pay Amt	Prev Unpaid Amt
$110,893.00	$608.80	23JN-413T	7/8/2023	$1,950.00	$21,507.00	$1,950.00

Total Amount Due Including This Invoice: **$21,602.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington Hills, MI 48331

The following is a summary of time and expenses for professional services rendered on this
Matter for the period: July 1, 2023 to July 31, 2023

Fees by Staff:

Name	Hours	Rate	Amount
Joshua A. Siegel	9.3	$390.00	$3,627.00
Anne D. Ackerman	30.5	$300.00	$9,150.00
Aparna V. Kaliappan	27.5	$250.00	$6,875.00
Total Hours and Fees	**67.3**		**$19,652.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington Hills, MI 48331

*Invoice #:*23JL-413T
Date: 08/04/2023
Invoice Due Date: 09/18/2023
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For professional services rendered for the period July 1, 2023 to July 31, 2023

Date	Staff	Description	Hrs
07/05/2023	AVK	Review and sign protective order	0.5
07/05/2023	JAS	Configure and test relativity access	0.5
07/06/2023	JAS	Revise and edit Bosch audit report	0.4
07/07/2023	JAS	Configure and test relativity access	0.3
07/07/2023	JAS	Revise and edit Bosch audit report	3.7
07/07/2023	AVK	Extract and consolidate document metadata from Relativity	1.2
07/07/2023	AVK	High-level review of produced documents in Relativity	1.2
07/07/2023	AVK	Review draft audit findings report	1.5
07/10/2023	AVK	Perform keyword searches and review produced documents in Relativity	2.5
07/10/2023	ADA	Research Relativity email production	2.0
07/11/2023	AVK	Review pleadings and documents	2.7
07/11/2023	ADA	Research and review email communication for relevant issues	2.4
07/13/2023	AVK	Review Relativity documents re: developer notes and initial project tasks	2.3
07/13/2023	ADA	Review pleadings re: allegations and audit report	2.4
07/13/2023	AVK	Review Relativity documents re: JSON files	1.7
07/14/2023	ADA	Review Relativity documents re: CDR	1.8
07/14/2023	ADA	Review Relativity documents re: Reverse Engineering	2.1
07/14/2023	ADA	Review Relativity documents re: CDR Replay	2.2
07/14/2023	AVK	Review Relativity documents re: JSON files	2.6
07/17/2023	AVK	Review emails and attachments from May 2017 in Relativity	1.8
07/17/2023	ADA	Review Relativity documents re crash scan	1.8
07/17/2023	ADA	Review Relativity documents re AutoCDR	2.1
07/17/2023	ADA	Review Relativity documents re development	1.5
07/18/2023	ADA	Review Relativity documents	2.8
07/18/2023	AVK	Review emails and attachments in Relativity	1.0
07/18/2023	ADA	Review and analyze email communication	3.2
07/18/2023	JAS	Review and address counsel feedback	0.3
07/19/2023	AVK	Review emails and attachments in Relativity	0.5
07/20/2023	ADA	Review Relativity email docouments re busmaster	1.9
07/20/2023	ADA	Review Relativity email documents	2.1
07/24/2023	ADA	Review Relativity emails and attachments	2.2
07/25/2023	AVK	Review emails and attachments in Relativity	1.3

Date		Description	Hours
07/26/2023	JAS	Review and Revise Audit report	1.3
07/26/2023	AVK	Review technical emails and attachments in Relativity	2.9
07/27/2023	AVK	Review emails and attachments from May 2017 in Relativity	2.6
07/27/2023	JAS	Review and Revise Audit report	2.1
07/27/2023	AVK	Review technical emails and attachments in Relativity	1.2
07/31/2023	JAS	Revise and finalize report	0.7
		Total Hours for Period	**67.3**

DisputeSoft
Software Dispute Experts

6116 Executive Blvd., Suite 330, North Bethesda, MD 20852
main **301.251.6313** · cell **518.982.7344** · fax **240.465.4442**
rsubbu@disputesoft.com
disputesoft.com

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

*Invoice #:*22AP-413T
Date: 05/04/2022
Invoice Due Date: 06/18/2022
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For Professional Services Rendered:

Period: April 1, 2022 to April 30, 2022

Total Fees	$2,472.00
Total Expenses	$0.00
New Charges:	$2,472.00
Outstanding Balance:	
New Balance:	**$2,472.00**

Payment by ACH/Wire:		Payment by Check:
Beneficiary Name:	Taron, LLC dba DisputeSoft	Taron, LLC dba DisputeSoft
Bank SWIFT - Name:	KEYBUS33 - KeyBank N. A.	25 Cranbrook
Bank Routing:	021300077	Avon, CT 06001
Account Number:	776871001553	USA

Account Summary

Services BTD	Expenses BTD	Last Inv Num	Last Inv Date	Last Inv Amt	Last Pay Amt	Prev Unpaid Amt
$2,472.00	$0.00	--	--	$0.00	$0.00	$0.00

Total Amount Due Including This Invoice: **$2,472.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

The following is a summary of time and expenses for professional services rendered on this
Matter for the period: April 1, 2022 to April 30, 2022

Fees by Staff:

Name	Hours	Rate	Amount
Raj Subbu	1.2	$500.00	$600.00
Joshua A. Siegel	4.8	$390.00	$1,872.00
Total Hours and Fees	**6.0**		**$2,472.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

For professional services rendered for the period April 1, 2022 to April 30, 2022

Date	Staff	Description	Hrs
04/21/2022	JAS	Introductory call with counsel	1.2
04/21/2022	JAS	Review Audit Protocol in preparation for call, review post call notes	1.3
04/21/2022	RS	Kickoff call with Bosch and Dykema counsels, and Bill Rose @ Bosch	1.2
04/28/2022	JAS	Consolidate notes from calls with counsel, flow of information	0.7
04/28/2022	JAS	Review and answer counsel questions	0.5
04/28/2022	JAS	Call with counsel	1.1
		Total Hours for Period	**6.0**

DisputeSoft
Software Dispute Experts

6116 Executive Blvd., Suite 330, North Bethesda, MD 20852
main **301.251.6313** · cell **518.982.7344** · fax **240.465.4442**
rsubbu@disputesoft.com
disputesoft.com

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

Invoice #: 22JL-413T
Date: 08/04/2022
Invoice Due Date: 09/18/2022
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For Professional Services Rendered:

Period: July 1, 2022 to July 31, 2022

Total Fees	$3,744.00
Total Expenses	$0.00
New Charges:	$3,744.00
Outstanding Balance:	$117.00
New Balance:	**$3,861.00**

Payment by ACH/Wire:

Beneficiary Name:	Taron, LLC dba DisputeSoft
Bank SWIFT - Name:	KEYBUS33 - KeyBank N. A.
Bank Routing:	021300077
Account Number:	776871001553

Account Summary						
Services BTD	Expenses BTD	Last Inv Num	Last Inv Date	Last Inv Amt	Last Pay Amt	Prev Unpaid Amt
$8,244.00	$210.94	22JN-413T	7/8/2022	$117.00	$2,121.94	$117.00

Total Amount Due Including This Invoice: **$3,861.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

The following is a summary of time and expenses for professional services rendered on this
Matter for the period: July 1, 2022 to July 31, 2022

Fees by Staff:

Name	Hours	Rate	Amount
Joshua A. Siegel	9.6	$390.00	$3,744.00
Total Hours and Fees	**9.6**		**$3,744.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

*Invoice #:*22JL-413T
Date: 08/04/2022
Invoice Due Date: 09/18/2022
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For professional services rendered for the period July 1, 2022 to July 31, 2022

Date	Staff	Description	Hrs
07/12/2022	JAS	Draft potential production alternative methods for audit	0.5
07/12/2022	JAS	Review proposed code keyword listing	0.8
07/12/2022	JAS	Call with counsel	0.4
07/13/2022	JAS	Review proposed code keyword listing	1.1
07/13/2022	JAS	Test and resolve dtSearch installation issues	0.5
07/20/2022	JAS	Review and respond to counsel questions	0.4
07/22/2022	JAS	Review and respond to counsel questions	0.8
07/25/2022	JAS	Test and track installation files and registry entries	1.8
07/26/2022	JAS	Test and track installation files and registry entries	2.5
07/27/2022	JAS	Research remote software and potential issues	0.8

Total Hours for Period	**9.6**

DisputeSoft
Software Dispute Experts

6116 Executive Blvd., Suite 330, North Bethesda, MD 20852
main **301.251.6313** · cell **518.982.7344** · fax **240.465.4442**
rsubbu@disputesoft.com
disputesoft.com

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

Invoice #: 22JN-413T
Date: 07/08/2022
Invoice Due Date: 08/22/2022
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For Professional Services Rendered:

Period: June 1, 2022 to June 30, 2022

Total Fees	$117.00
Total Expenses	$0.00
New Charges:	$117.00
Outstanding Balance:	$0.00
New Balance:	**$117.00**

Payment by ACH/Wire:

Beneficiary Name:	Taron, LLC dba DisputeSoft
Bank SWIFT - Name:	KEYBUS33 - KeyBank N. A.
Bank Routing:	021300077
Account Number:	776871001553

Account Summary

Services BTD	Expenses BTD	Last Inv Num	Last Inv Date	Last Inv Amt	Last Pay Amt	Prev Unpaid Amt
$4,500.00	$210.94	22MY-413T	6/8/2022	$2,121.94	$2,121.94	$0.00

Total Amount Due Including This Invoice: **$117.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

*Invoice #:*22JN-413T
Date: 07/08/2022
Invoice Due Date: 08/22/2022
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

The following is a summary of time and expenses for professional services rendered on this
Matter for the period: June 1, 2022 to June 30, 2022

Fees by Staff:

Name	Hours	Rate	Amount
Joshua A. Siegel	0.3	$390.00	$117.00
Total Hours and Fees	**0.3**		**$117.00**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

For professional services rendered for the period June 1, 2022 to June 30, 2022

Date	Staff	Description	Hrs
06/01/2022	JAS	Review new audit reports from counsel	0.3
		Total Hours for Period	**0.3**

DisputeSoft
Software Dispute Experts

6116 Executive Blvd., Suite 330, North Bethesda, MD 20852
main **301.251.6313** · cell **518.982.7344** · fax **240.465.4442**
rsubbu@disputesoft.com
disputesoft.com

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

Invoice #: 22MY-413T
Date: 06/08/2022
Invoice Due Date: 07/23/2022
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For Professional Services Rendered:

Period: May 1, 2022 to May 31, 2022

Total Fees	$1,911.00
Total Expenses	$210.94
New Charges:	$2,121.94
Outstanding Balance:	$2,472.00
New Balance:	**$4,593.94**

Payment by ACH/Wire:	
Beneficiary Name:	Taron, LLC dba DisputeSoft
Bank SWIFT - Name:	KEYBUS33 - KeyBank N. A.
Bank Routing:	021300077
Account Number:	776871001553

Account Summary						
Services BTD	Expenses BTD	Last Inv Num	Last Inv Date	Last Inv Amt	Last Pay Amt	Prev Unpaid Amt
$4,383.00	$210.94	22AP-413T	5/4/2022	$2,472.00	$0.00	$2,472.00

Total Amount Due Including This Invoice: $4,593.94

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

The following is a summary of time and expenses for professional services rendered on this Matter for the period: May 1, 2022 to May 31, 2022

Fees by Staff:

Name	Hours	Rate	Amount
Joshua A. Siegel	4.9	$390.00	$1,911.00
Total Hours and Fees	**4.9**		**$1,911.00**

Expenses by Staff:

Name	Amount
Joshua A. Siegel	$210.94
Total Out-of-Pocket Expenses	**$210.94**

Steve Hurvitz, Esq.
Robert Bosch LLC
38000 Hills Tech Drive
Farmington HIlls, MI 48331

*Invoice #:*22MY-413T
Date: 06/08/2022
Invoice Due Date: 07/23/2022
Matter Name: Bosch v. Collision Sciences

Tax Identification No.: 86-2563576

For professional services rendered for the period May 1, 2022 to May 31, 2022

Date	Staff	Description	Hrs
05/09/2022	JAS	Review notes and create diagram for high level overview	0.7
05/09/2022	JAS	Call with counsel	0.9
05/11/2022	JAS	Review and provide responses to Collision Science descriptions	1.3
05/25/2022	JAS	Call with counsel: Audit coordination	0.3
05/25/2022	JAS	Review Collision Sciences responses to software operation questions	0.3
05/26/2022	JAS	Review materials and build keyword listing	1.4
		Total Hours for Period	**4.9**

Expenses for the period: May 1, 2022 to May 31, 2022

Date	Staff	Description	Amount
05/25/2022	JAS	dtSearch Desktop - single user license	$210.94
		Total Expenses for Period	**$210.94**

Tax Invoice

Invoice No: BI49640244
Date: May 25, 2022
Order No: 182712095

Provided by:

Avangate Inc dba 2Checkout
11700 Great Oaks Way no. 210, 30022
Alpharetta , Georgia
Country VAT ID: United States of America
VAT ID: 26-0160456
E-mail: info@2checkout.com

Provided to:

DisputeSoft
Joshua HelfinSiegel
6116 Executive Boulevard
North Bethesda 20852
Maryland, United States of America
E-mail: jsiegel@disputesoft.com

Delivered to:

DisputeSoft
Joshua HelfinSiegel
6116 Executive Boulevard
North Bethesda 20852
Maryland, United States of America
E-mail: jsiegel@disputesoft.com

Invoice Details

No	Products	Units	Unit Price (USD)	Sales TAX (USD)	Value (USD)
1	dtSearch Desktop - single user license	1	199.00	11.94 (6.00 %)	199.00

Delivery date: 2022-05-25
Sales TAX (USD): 11.94
Total (USD): 210.94
Invoice Status: PAID

Payment Details

Payment method used: American Express
Your credit card (###########2019) has been successfully charged. Please note that the charge on your credit card will appear as 'dtsearch.com'.

The purchased products and services are delivered in accordance with the terms and conditions published on www.dtsearch.com website and agreed during the online order. For technical support please contact support@dtsearch.com

2checkout

Taron, LLC dba DisputeSoft

6116 Executive Boulevard, Suite 330

North Bethesda, MD 20852

Tel: 301-251-6313 Fax: 240-465-4442

Filters Used:
- Time Expense Date: 1/1/2022 to 7/31/2023
- Time Expense Matter ID: 413T Bosch v Collision Sciences: to 413T Bosch v Collision Sciences:

Law Firm ID: Dykema Gossett PLLC - *Dykema Gossett PLLC*

` = *Invoiced (mouse over for #), * = Marked as Billed*

Date	Description	Hrs	B-Hr/Unit	Bill Rate	Amount	*

Matter ID - Name (Manager): 413T Bosch v Collision Sciences: - *Bosch v. Collision Sciences* **(Joshua Siegel)**

<u>Services:</u>

Employee

ADA - *Anne D. Ackerman*

Date	Description	Hrs	B-Hr/Unit	Bill Rate	Amount	*
5/11/2023	Assist with review of citations of draft expert report	1.60	1.60	$300.00	$480.00	
5/11/2023	Assist with review and edit of draft expert report	2.70	2.70	$300.00	$810.00	
7/10/2023	Research Relativity email production	2.00	2.00	$300.00	$600.00	
7/11/2023	Research and review email communication for relevant issues	2.40	2.40	$300.00	$720.00	
7/13/2023	Review pleadings re: allegations and audit report	2.40	2.40	$300.00	$720.00	
7/14/2023	Review Relativity documents re: CDR Replay	2.20	2.20	$300.00	$660.00	
7/14/2023	Review Relativity documents re: Reverse Engineering	2.10	2.10	$300.00	$630.00	
7/14/2023	Review Relativity documents re: CDR	1.80	1.80	$300.00	$540.00	
7/17/2023	Review Relativity documents re crash scan	1.80	1.80	$300.00	$540.00	
7/17/2023	Review Relativity documents re AutoCDR	2.10	2.10	$300.00	$630.00	
7/17/2023	Review Relativity documents re development	1.50	1.50	$300.00	$450.00	
7/18/2023	Review Relativity documents	2.80	2.80	$300.00	$840.00	
7/18/2023	Review and analyze email communication	3.20	3.20	$300.00	$960.00	
7/20/2023	Review Relativity email docouments re busmaster	1.90	1.90	$300.00	$570.00	
7/20/2023	Review Relativity email documents	2.10	2.10	$300.00	$630.00	
7/24/2023	Review Relativity emails and attachments	2.20	2.20	$300.00	$660.00	
	ADA Total:	**34.80**	**34.80**		**$10,440.00**	

AVK - *Aparna V. Kaliappan*

Date	Description	Hrs	B-Hr/Unit	Bill Rate	Amount	*
7/5/2023	Review and sign protective order	0.50	0.50	$250.00	$125.00	
7/7/2023	Extract and consolidate document metadata from Relativity	1.20	1.20	$250.00	$300.00	
7/7/2023	Review draft audit findings report	1.50	1.50	$250.00	$375.00	
7/7/2023	High-level review of produced documents in Relativity	1.20	1.20	$250.00	$300.00	
7/10/2023	Perform keyword searches and review produced documents in Relativity	2.50	2.50	$250.00	$625.00	
7/11/2023	Review pleadings and documents	2.70	2.70	$250.00	$675.00	
7/13/2023	Review Relativity documents re: developer notes and initial project tasks	2.30	2.30	$250.00	$575.00	
7/13/2023	Review Relativity documents re: JSON files	1.70	1.70	$250.00	$425.00	
7/14/2023	Review Relativity documents re: JSON files	2.60	2.60	$250.00	$650.00	
7/17/2023	Review emails and attachments from May 2017 in Relativity	1.80	1.80	$250.00	$450.00	
7/18/2023	Review emails and attachments in Relativity	1.00	1.00	$250.00	$250.00	
7/19/2023	Review emails and attachments in Relativity	0.50	0.50	$250.00	$125.00	
7/25/2023	Review emails and attachments in Relativity	1.30	1.30	$250.00	$325.00	
7/26/2023	Review technical emails and attachments in Relativity	2.90	2.90	$250.00	$725.00	
7/27/2023	Review emails and attachments from May 2017 in Relativity	2.60	2.60	$250.00	$650.00	
7/27/2023	Review technical emails and attachments in Relativity	1.20	1.20	$250.00	$300.00	
	AVK Total:	**27.50**	**27.50**		**$6,875.00**	

JAS - *Joshua A. Siegel*

Date	Description	Hrs	B-Hr/Unit	Bill Rate	Amount	*
4/21/2022	Review Audit Protocol in preparation for call, review post call notes	1.30	1.30	$390.00	$507.00	
4/21/2022	Introductory call with counsel	1.20	1.20	$390.00	$468.00	
4/28/2022	Call with counsel	1.10	1.10	$390.00	$429.00	
4/28/2022	Review and answer counsel questions	0.50	0.50	$390.00	$195.00	

Matter ID - Name (Manager): 413T Bosch v Collision Sciences: - *Bosch v. Collision Sciences* **(Joshua Siegel)**

JAS - *Joshua A. Siegel*

Date	Description	Hrs	B-Hr/Unit	Bill Rate	Amount	*
4/28/2022	Consolidate notes from calls with counsel, flow of information	0.70	0.70	$390.00	$273.00	
5/9/2022	Review notes and create diagram for high level overview	0.70	0.70	$390.00	$273.00	
5/9/2022	Call with counsel	0.90	0.90	$390.00	$351.00	

Date	Description	Hours	Hours	Rate	Amount
5/11/2022	Review and provide responses to Collision Science descriptions	1.30	1.30	$390.00	$507.00
5/25/2022	Call with counsel: Audit coordination	0.30	0.30	$390.00	$117.00
5/25/2022	Review Collision Sciences responses to software operation questions	0.30	0.30	$390.00	$117.00
5/26/2022	Review materials and build keyword listing	1.40	1.40	$390.00	$546.00
6/1/2022	Review new audit reports from counsel	0.30	0.30	$390.00	$117.00
7/12/2022	Draft potential production alternative methods for audit	0.50	0.50	$390.00	$195.00
7/12/2022	Review proposed code keyword listing	0.80	0.80	$390.00	$312.00
7/12/2022	Call with counsel	0.40	0.40	$390.00	$156.00
7/13/2022	Review proposed code keyword listing	1.10	1.10	$390.00	$429.00
7/13/2022	Test and resolve dtSearch installation issues	0.50	0.50	$390.00	$195.00
7/20/2022	Review and respond to counsel questions	0.40	0.40	$390.00	$156.00
7/22/2022	Review and respond to counsel questions	0.80	0.80	$390.00	$312.00
7/25/2022	Test and track installation files and registry entries	1.80	1.80	$390.00	$702.00
7/26/2022	Test and track installation files and registry entries	2.50	2.50	$390.00	$975.00
7/27/2022	Research remote software and potential issues	0.80	0.80	$390.00	$312.00
8/2/2022	Draft TeamViewer setup directions	1.10	1.10	$390.00	$429.00
8/3/2022	Research viable teamviewer alternatives and potential	0.80	0.80	$390.00	$312.00
8/9/2022	Setup and test screen recording software, configure review computer, encrypt and test external drive	1.30	1.30	$390.00	$507.00
8/10/2022	Setup and test screen recording software, configure review computer, encrypt and test external drive	1.50	1.50	$390.00	$585.00
8/11/2022	Setup and test screen recording software, configure review computer, encrypt and test external drive	0.50	0.50	$390.00	$195.00
8/15/2022	Setup and test screen recording software, configure review computer, encrypt and test external drive	2.80	2.80	$390.00	$1,092.00
8/15/2022	Connect to audit computer, begin indexing	0.30	0.30	$390.00	$117.00
8/16/2022	Setup and test screen recording software, configure review computer, encrypt and test external drive	0.70	0.70	$390.00	$273.00
8/16/2022	Audit of computer	4.90	4.90	$390.00	$1,911.00
8/17/2022	Setup and test screen recording software, configure review computer, encrypt and test external drive	0.50	0.50	$390.00	$195.00
8/17/2022	Review, update, and consolidate audit notes	0.40	0.40	$390.00	$156.00
8/17/2022	Audit of computer	3.60	3.60	$390.00	$1,404.00
8/18/2022	Setup and test screen recording software, configure review computer, encrypt and test external drive	0.50	0.50	$390.00	$195.00
8/18/2022	Review, update, and consolidate audit notes	0.60	0.60	$390.00	$234.00
8/18/2022	Audit of computer	4.50	4.50	$390.00	$1,755.00
8/19/2022	Audit of computer	4.50	4.50	$390.00	$1,755.00
8/19/2022	Review, update, and consolidate audit notes	2.10	2.10	$390.00	$819.00
8/19/2022	Setup and test screen recording software, configure review computer, encrypt and test external drive	0.30	0.30	$390.00	$117.00
8/22/2022	Call with counsel for status updates	1.50	1.50	$390.00	$585.00
8/22/2022	Review, update, and consolidate audit notes	4.10	4.10	$390.00	$1,599.00

Matter ID - Name (Manager): 413T Bosch v Collision Sciences: - *Bosch v. Collision Sciences* **(Joshua Siegel)**

JAS - *Joshua A. Siegel*

Date	Description	Hours	Hours	Rate	Amount
8/24/2022	Audit of computer	0.30	0.30	$390.00	$117.00
8/24/2022	Review, update, and consolidate audit notes	1.50	1.50	$390.00	$585.00
8/25/2022	Audit of computer	3.70	3.70	$390.00	$1,443.00
8/25/2022	Review, update, and consolidate audit notes	0.80	0.80	$390.00	$312.00
8/26/2022	Review, update, and consolidate audit notes	0.60	0.60	$390.00	$234.00
8/26/2022	Audit of computer	0.80	0.80	$390.00	$312.00
8/29/2022	Review, update, and consolidate audit notes	0.60	0.60	$390.00	$234.00
9/12/2022	Call with counsel re: Audit	0.30	0.30	$390.00	$117.00
9/16/2022	Draft and revise declaration	6.00	6.00	$390.00	$2,340.00
9/19/2022	Draft and revise declaration	2.80	2.80	$390.00	$1,092.00
10/11/2022	Setup and test new BitBucket access	0.40	0.40	$390.00	$156.00
10/14/2022	Answer counsel questions	0.50	0.50	$390.00	$195.00
10/21/2022	Assist counsel with search suggestion terms and areas	0.60	0.60	$390.00	$234.00
11/28/2022	Bitbucket Audit	1.10	1.10	$390.00	$429.00
11/28/2022	Prep system for bitbucket audit, setup initial document, review results and summarize info	1.50	1.50	$390.00	$585.00
11/29/2022	Prep system for bitbucket audit, setup initial document, review results and summarize info	1.10	1.10	$390.00	$429.00
11/29/2022	Bitbucket Audit	0.80	0.80	$390.00	$312.00
11/30/2022	Prep system for bitbucket audit, setup initial document, review	0.70	0.70	$390.00	$273.00

	results and summarize into				
12/1/2022	Prep system for bitbucket audit, setup initial document, review results and summarize info	0.50	0.50	$390.00	$195.00
12/1/2022	Bitbucket Audit	0.90	0.90	$390.00	$351.00
12/2/2022	Prep system for bitbucket audit, setup initial document, review results and summarize info	0.70	0.70	$390.00	$273.00
12/2/2022	Bitbucket Audit	2.30	2.30	$390.00	$897.00
12/5/2022	Prep system for bitbucket audit, setup initial document, review results and summarize info	0.50	0.50	$390.00	$195.00
12/6/2022	Call with counsel	1.10	1.10	$390.00	$429.00
12/6/2022	Prep system for bitbucket audit, setup initial document, review results and summarize info	0.30	0.30	$390.00	$117.00
12/7/2022	Setup initial document for AWS audit, review results and summarize info	0.50	0.50	$390.00	$195.00
12/7/2022	Amazon Web Services and RDS Audit	1.40	1.40	$390.00	$546.00
12/8/2022	Researching AWS Elastic Beanstalk, VPC, EC2, and RDS options and connectivity	1.80	1.80	$390.00	$702.00
12/8/2022	Amazon Web Services and RDS Audit	1.80	1.80	$390.00	$702.00
12/8/2022	Troubleshooting AWS database connection	1.10	1.10	$390.00	$429.00
12/9/2022	Researching AWS Elastic Beanstalk, VPC, EC2, and RDS options and connectivity	0.60	0.60	$390.00	$234.00
12/9/2022	Troubleshooting AWS database connection	0.70	0.70	$390.00	$273.00
12/9/2022	Review and research audit data in re: replay source	1.10	1.10	$390.00	$429.00
12/12/2022	Troubleshooting AWS database connection	0.30	0.30	$390.00	$117.00
12/12/2022	Review and research audit data in re: replay source	1.40	1.40	$390.00	$546.00
12/12/2022	Amazon Web Services and RDS Audit	0.70	0.70	$390.00	$273.00
12/14/2022	Review and research audit data in re: replay source	1.00	1.00	$390.00	$390.00

Matter ID - Name (Manager): 413T Bosch v Collision Sciences: - *Bosch v. Collision Sciences* **(Joshua Siegel)**

JAS - *Joshua A. Siegel*

12/15/2022	Review and research audit data in re: replay source	0.40	0.40	$390.00	$156.00
12/21/2022	Source code repository audit: CDR data	0.80	0.80	$390.00	$312.00
12/21/2022	Review and research audit data in re: replay source	0.70	0.70	$390.00	$273.00
12/22/2022	Review and research audit data in re: replay source	0.80	0.80	$390.00	$312.00
12/22/2022	Source code repository audit: CDR data	1.50	1.50	$390.00	$585.00
12/27/2022	Source code repository audit: CDR data	0.30	0.30	$390.00	$117.00
12/28/2022	Drafting Bosch Audit Report	3.30	3.30	$390.00	$1,287.00
12/29/2022	Drafting Bosch Audit Report	1.00	1.00	$390.00	$390.00
1/4/2023	Draft Audit Findings Report	1.30	1.30	$390.00	$507.00
1/5/2023	Draft Audit Findings Report	0.70	0.70	$390.00	$273.00
1/6/2023	Draft Audit Findings Report	1.20	1.20	$390.00	$468.00
1/9/2023	Draft Audit Findings Report	2.20	2.20	$390.00	$858.00
1/13/2023	Draft Audit Findings Report	3.20	3.20	$390.00	$1,248.00
1/20/2023	Draft audit findings report	2.50	2.50	$390.00	$975.00
1/23/2023	Review extension listing for new production prior to call with counsel	0.50	0.50	$390.00	$195.00
1/23/2023	Draft Audit Findings Report	0.80	0.80	$390.00	$312.00
1/23/2023	Call with Counsel	0.90	0.90	$390.00	$351.00
1/24/2023	Draft Audit Findings Report	1.00	1.00	$390.00	$390.00
1/25/2023	Review extension listing for new production prior to call with counsel	1.10	1.10	$390.00	$429.00
1/26/2023	Draft Audit Findings Report	1.40	1.40	$390.00	$546.00
1/26/2023	Review new EULAs and terms	0.70	0.70	$390.00	$273.00
1/26/2023	Review extension listing for new production prior to call with counsel	0.60	0.60	$390.00	$234.00
1/27/2023	Draft Audit Findings Report	4.60	4.60	$390.00	$1,794.00
2/10/2023	Review new production re: Reverse Engineering	1.10	1.10	$390.00	$429.00
2/14/2023	Review new production documents	0.30	0.30	$390.00	$117.00
3/21/2023	Call with counsel to review Audit findings report	0.70	0.70	$390.00	$273.00
3/21/2023	Review recent documents to prepare for Bosch Call with	0.80	0.80	$390.00	$312.00
3/22/2023	Drafting Audit report	0.50	0.50	$390.00	$195.00
3/30/2023	Draft audit findings report	3.80	3.80	$390.00	$1,482.00
3/31/2023	Draft audit findings report	0.50	0.50	$390.00	$195.00
4/3/2023	Draft Audit Findings Report	0.60	0.60	$390.00	$234.00
4/4/2023	Draft Audit Findings Report	2.80	2.80	$390.00	$1,092.00
4/6/2023	Draft Audit Findings Report	0.50	0.50	$390.00	$195.00
4/7/2023	Draft Audit Findings Report	3.40	3.40	$390.00	$1,326.00

Date	Description			Rate	Amount
4/11/2023	Draft Bosch Audit report	2.30	2.30	$390.00	$897.00
4/12/2023	Draft Bosch Audit report	0.70	0.70	$390.00	$273.00
4/13/2023	Draft Bosch Audit report	2.10	2.10	$390.00	$819.00
4/14/2023	Draft Bosch Audit report	3.40	3.40	$390.00	$1,326.00
4/17/2023	Draft Bosch Audit report	2.50	2.50	$390.00	$975.00
4/18/2023	Draft Bosch Audit report	1.10	1.10	$390.00	$429.00
4/19/2023	Draft Bosch Audit report	2.70	2.70	$390.00	$1,053.00
4/20/2023	Draft Bosch Audit report	1.20	1.20	$390.00	$468.00
4/21/2023	Draft Bosch Audit report	3.20	3.20	$390.00	$1,248.00

Matter ID - Name (Manager): 413T Bosch v Collision Sciences: - *Bosch v. Collision Sciences* **(Joshua Siegel)**

JAS - *Joshua A. Siegel*

Date	Description			Rate	Amount
4/27/2023	Draft and update Audit Report	1.30	1.30	$390.00	$507.00
4/28/2023	Draft and update Audit Report	2.70	2.70	$390.00	$1,053.00
5/1/2023	Draft and update Audit Report	2.20	2.20	$390.00	$858.00
5/2/2023	Draft and update Audit Report	1.70	1.70	$390.00	$663.00
5/3/2023	Draft and update Audit Report	1.30	1.30	$390.00	$507.00
5/4/2023	Draft and update Audit Report	3.30	3.30	$390.00	$1,287.00
5/5/2023	Draft and update Audit Report	3.70	3.70	$390.00	$1,443.00
5/8/2023	Draft and update Audit Report	2.60	2.60	$390.00	$1,014.00
5/9/2023	Draft and update Audit Report	4.10	4.10	$390.00	$1,599.00
5/10/2023	Draft and update Audit Report	5.10	5.10	$390.00	$1,989.00
5/11/2023	Draft and update Audit Report	2.30	2.30	$390.00	$897.00
5/12/2023	Draft and update Audit Report	3.10	3.10	$390.00	$1,209.00
5/15/2023	Draft and update Audit Report	2.10	2.10	$390.00	$819.00
5/16/2023	Draft and update Audit Report	1.00	1.00	$390.00	$390.00
5/22/2023	Review new production	0.50	0.50	$390.00	$195.00
5/22/2023	Draft Audit report	0.50	0.50	$390.00	$195.00
5/23/2023	Review new production	1.10	1.10	$390.00	$429.00
5/23/2023	Draft Audit report	0.80	0.80	$390.00	$312.00
5/23/2023	Call with counsel for status update	1.70	1.70	$390.00	$663.00
5/24/2023	Draft Audit report	2.30	2.30	$390.00	$897.00
5/25/2023	Draft Audit report	2.70	2.70	$390.00	$1,053.00
5/26/2023	Draft Audit report	3.30	3.30	$390.00	$1,287.00
5/26/2023	Validate software version release dates	0.40	0.40	$390.00	$156.00
5/26/2023	Create and format report Attachments	2.10	2.10	$390.00	$819.00
5/26/2023	Review new production	1.40	1.40	$390.00	$546.00
6/19/2023	Review, revise and edit Audit report	1.10	1.10	$390.00	$429.00
6/19/2023	Encrypted email research and setup per Leone Apollonia	1.70	1.70	$390.00	$663.00
6/20/2023	Review, revise and edit Audit report	0.30	0.30	$390.00	$117.00
6/26/2023	Revise and update Audit report	0.80	0.80	$390.00	$312.00
6/26/2023	Case status updates for counsel	0.60	0.60	$390.00	$234.00
6/27/2023	Revise and update Audit report	0.50	0.50	$390.00	$195.00
7/5/2023	Configure and test relativity access	0.50	0.50	$390.00	$195.00
7/6/2023	Revise and edit Bosch audit report	0.40	0.40	$390.00	$156.00
7/7/2023	Configure and test relativity access	0.30	0.30	$390.00	$117.00
7/7/2023	Revise and edit Bosch audit report	3.70	3.70	$390.00	$1,443.00
7/18/2023	Review and address counsel feedback	0.30	0.30	$390.00	$117.00
7/26/2023	Review and Revise Audit report	1.30	1.30	$390.00	$507.00
7/27/2023	Review and Revise Audit report	2.10	2.10	$390.00	$819.00
7/31/2023	Revise and finalize report	0.70	0.70	$390.00	$273.00
	JAS Total:	**232.20**	**232.20**		**$90,558.00**

RS - *Raj Subbu*

Date	Description			Rate	Amount
4/21/2022	Kickoff call with Bosch and Dykema counsels, and Bill Rose @ Bosch	1.20	1.20	$500.00	$600.00
	RS Total:	**1.20**	**1.20**		**$600.00**

TJW - *Thomas J. Wolf*

Matter ID - Name (Manager): 413T Bosch v Collision Sciences: - *Bosch v. Collision Sciences* **(Joshua Siegel)**

TJW - *Thomas J. Wolf*

Date	Description			Rate	Amount
12/7/2022	Review and sign protective order	0.60	0.60	$275.00	$165.00
2/15/2023	Review and revise draft of audit report	1.60	1.60	$275.00	$440.00
2/15/2023	Review and revise draft of J.Siegel audit report	1.80	1.80	$275.00	$495.00
2/16/2023	Review and revise draft of J.Siegel audit report	0.80	0.80	$275.00	$220.00
2/17/2023	Review and revise draft of J.Siegel audit report	0.40	0.40	$275.00	$110.00
5/10/2023	Review and revise draft of J.Siegel software audit report	1.90	1.90	$275.00	$522.50

Date	Description					
5/10/2023	Assist with review and revising draft of audit report	1.70	1.70	$275.00	$467.50	
	TJW Total:	**8.80**	**8.80**		**$2,420.00**	
	Employee Total:	304.50	304.50		$110,893.00	
	Services Total:	304.50	304.50		$110,893.00	

Expenses:

Employee

JAS - *Joshua A. Siegel*

Date	Description			
5/25/2022	dtSearch Desktop - single user license		1.00	$210.94
8/15/2022	Team Viewer		1.00	$421.74
	JAS Total:		**2.00**	**$632.68**

RS - *Raj Subbu*

Date	Description			
10/12/2022	Refund of MD Sales Tax on Commercial Use of TeamViewer Purchased 8/15/22 - MD Senate Bill 723		1.00	($23.88)
	RS Total:		**1.00**	**($23.88)**
	Employee Total:		**3.00**	**$608.80**
	Expenses Total:		**3.00**	**$608.80**
	Matter 413T Bosch v Collision Sciences: Total:	**304.50**	**307.50**	**$111,501.80**

Matter Summary for *Bosch v. Collision Sciences* **(**413T Bosch v Collision Sciences:**)**

	Billable	Non Billable	Total
Services	$110,893.00	$0.00	$110,893.00
Extra Services	$0.00	$0.00	$0.00
Total Services	**$110,893.00**	**$0.00**	**$110,893.00**
Expenses	$608.80	$0.00	$608.80
Extra Expenses	$0.00	$0.00	$0.00
Total Expense	**$608.80**	**$0.00**	**$608.80**
Grand Total	**$111,501.80**	**$0.00**	**$111,501.80**

Grand Total Time:	304.50	$110,893.00
Grand Total Expenses:	3.00	$608.80
Grand Total Time & Expenses:		$111,501.80

Web Suite Standard Report Copyright © 2024 BQE Software, Inc.

From: Jason from Collision Sciences [jbayley@collisionsciences.ca]
on behalf of Jason from Collision Sciences <jbayley@collisionsciences.ca> [jbayley@collisionsciences.ca]
Sent: 9/12/2017 12:38:23 AM
To: Brian Chang-Yun Hsu [changyun.hsu@gmail.com]
Subject: Re: Vector Product Assessment

Brian,

Can you please forward me your resume so I have it on file?

Also, as a recap to our call:
1. Tentative Th. meetup for demo of code/tools (next week okay too)
2. Help determine req'd CANalyzer tool (used or what we need based on how we proceed, i.e. GM may or may not be a priority). I will forward my contact at Vector as well.
3. Will introduce you to Jonathan (full stack dev) via email (review all code first, make your own notes for questions) and we can create a picture of the project for you... and next steps. The main thing we need is to know is how the code functions (and why), and what CANID etc. are used (i need notes and ypur jelp to make sure we get correct and complete OEM info)
4. To think on: Can we automate the Reverse Engineering of many vehicles? Its unlikely, but i'm curious if its possible.

Lets work together to get you the equipment you need (I have a spare CANtact and field bosche kit you can learn to use right away). Bosche even uses Windows software... you'll need a new pc. I'm open to buying one for the company, but of you prefer to have it as personal, that's fine too.

I will come up with a template contract for the $20/hr (initial) rate, and that just covers std clauses and developed IP belonging to Collision Sciences. We can discuss timing/milestones for $30/hr soon.

Talk soon,

Jason

On Sep 11, 2017 8:19 PM, "Jason from Collision Sciences" <jbayley@collisionsciences.ca> wrote:
There are a couple of interesting articles by this company; here's one:

https://www.google.ca/url?sa=t&source=web&rct=j&url=https://www.ioactive.com/pdfs/IOActive_Adventures_in_Automotive_Networks_and_Control_Units.pdf&ved=0ahUKEwjnsd-3r57WAhVq6YMKHetwD_YQFgglMAA&usg=AFQjCNEDa97hZluvjE_6SPyf9FWPuQ49ng

On Sep 11, 2017 4:42 PM, "Brian Chang-Yun Hsu" <changyun.hsu@gmail.com> wrote:
Hi Jason,

I am free today (and now) if you would like to talk on the phone. Thanks!

Brian Hsu

Hrg. EX.
42

Brian Chang-Yun Hsu

On 11 September 2017 at 16:40, Jason from Collision Sciences <jbayley@collisionsciences.ca> wrote:
Thanks Brian. All is well...

I have to bring you up to speed on a few things. When would you like a quick talk on the phone?

J

On Sep 11, 2017 4:00 PM, "Brian Chang-Yun Hsu" <changyun.hsu@gmail.com> wrote:
Hi Jason,

Congratulations! I hope the mother and the baby are doing well.
I need to visit ServiceOntario on Friday, but am otherwise free to meet this week. Thanks!

Sincerely,

Brian Hsu

From:	Jason from Collision Sciences [jbayley@collisionsciences.ca]
on behalf of	Jason from Collision Sciences <jbayley@collisionsciences.ca> [jbayley@collisionsciences.ca]
Sent:	9/25/2017 2:03:00 PM
To:	Brian Chang-Yun Hsu [changyun.hsu@gmail.com]
CC:	Renan Pedrosa [renanjpedrosa@gmail.com]
Subject:	Re: RE: Crash Data Info (for Invent with Bosch)

I googled "reverse engineering encrypted firmware"....

https://reverseengineering.stackexchange.com/questions/14539/reverse-engineering-a-firmware-someone-can-help-me

http://jcjc-dev.com/2016/04/29/reversing-huawei-router-2-scouting-firmware/

On Mon, Sep 25, 2017 at 9:55 AM, Jason from Collision Sciences <jbayley@collisionsciences.ca> wrote:
Yes, we are still going to be getting a 2-day trial from RA Consulting.

Also, please don't give up too quickly on the firmware... maybe try the other software "firmwalker". Renan looked at the file and did say it looked binary.

Failing that, try and determine if it is in fact consistent with firmware or if we should try and pull the full firmware directly from the hardware (Attify tools?). Are there any other reverse engineering decryption techniques?

Renan, please share any progress/ideas on the firmware. You mentioned you had some other software reverse engineering ideas?

Jason

On Mon, Sep 25, 2017 at 9:38 AM, Brian Chang-Yun Hsu <changyun.hsu@gmail.com> wrote:
Hi everyone,

I just ran binwalk on those Bosch CDR firmware files with no luck. Binwalk does not produce any output at all from scanning those files. Maybe these S19 files are not really firmware files or Bosch has encrypted them.

When I installed binwalk, I did not have "git" on my Linux machine. Once I installed that, binwalk installed just fine.

I will spend some time today on the vehicle processes (vehicle.py) and see if VW has given us what we needed. Jason, can we still get that two-day free trial of the ODX Viewer from RA Consulting? The Tiny ODX Viewer works, but I am just curious to see if RAC has something better to offer.

Thanks!

Brian

Hrg. EX.
43

On 23 September 2017 at 11:20, Jason from Collision Sciences <jbayley@collisionsciences.ca> wrote:
Great update.

So, guys, have a look at the screenshot of the VW process (and/or code file vehicles.py).

I want to know if these files contain all the info we need to build out the VW process for expaning vehicle support. If so, I will arrange for more data. Im wondering now if this was available on ETI all along (they were not very helpful over there in helping me understand what was in their tek-net darabase!).

Im pretty sure there is no seed key for VW ... or most OEms... maybe in the future.

As far as I know from the initial dev, only Toyota required a seed key. We apparently did get Ford to work; though I also read in an online vehicle hack article frim IOactive that Ford has a seed key as well (maybe he didnt come across those vehicles yet).

Brian, good luck on the firmware analysis. What was the issue you resolved? It might help Renan. Keep notes on testing and progress!

Have a great weekend guys-

J

On Sep 22, 2017 11:23 PM, "Brian Chang-Yun Hsu" <changyun.hsu@gmail.com> wrote:
Thanks, Renan! I did not realize that the PDX was a zipped file and needed to be extracted. Now I get a bunch of .odx files which can be opened using the Tiny ODX Viewer.

These files contain A TON of information. I briefly looked through the airbag file and it even contains diagnostic trouble codes information. It seems that VW uses ISO 15765-2, which uses Extended Addressing. I found the CAN ID for airbag diagnostic request is 1813 and the response uses 1919 (this is for the Audi A1).

Regarding the firmware stuff, the creator of binwalk got back to me and I got binwalk to install just fine.

Have a good night!

Brian Chang-Yun Hsu

On 22 September 2017 at 22:34, Renan Pedrosa <renanjpedrosa@gmail.com> wrote:
Adding to my last email. For best reading results, you can use any XML viewer for the odx.

On Fri, Sep 22, 2017 at 10:27 PM, Renan Pedrosa <renanjpedrosa@gmail.com> wrote:
Hi Brian and Jason,

The pdx is a zipped package with several odx files. I used 7zip to extract the pdx file. The software PDXplorer did not work very well (freeware for pdx packages).

EL_ECMDFCC_001006.odx	07/09/2012 13:37	Arquivo ODX	3.447 KB
EL_LIBEPHVA_001030.odx	07/09/2012 13:37	Arquivo ODX	694 KB
EL_LIBImmob_001001.odx	07/09/2012 13:37	Arquivo ODX	53 KB
EV_AirbaUDS_007006.odx	07/09/2012 13:37	Arquivo ODX	350 KB
EV_CamerSysteRearView_JDS_003001.odx	07/09/2012 13:37	Arquivo ODX	2 KB
EV_ErglrContrModul1UDS_002056.odx	07/09/2012 13:37	Arquivo ODX	672 KB
EV_ErgirContrModul2UDS_002056.odx	07/09/2012 13:37	Arquivo ODX	673 KC
EV_HeadlRegulUDS_005141.ocx	07/09/2012 13:37	Arquivo ODX	3.9 KB
EV_ParkiAssis2UDS_003037.odx	07/09/2012 13:37	Arquivo ODX	6 KB
EV_TransContrModulUDS_003021.odx	07/09/2012 13:37	Arquivo ODX	173 KB
ES_DataLibra_001006.odx	07/09/2012 13:37	Arquivo ODX	430 KB
EV_Airba10AURE4_001014.odx	07/09/2012 13:37	Arquivo ODX	145 KB
EV_CamSysteCorneViewAUDI_004005.odx	07/09/2012 13:37	Arquivo ODX	320 KB
EV_ECM20TDI03L9C6019AE_A01105.odx	07/09/2012 13:37	Arquivo ODX	310 KB
EV_ECM20TDI03L9C6019AF_A01106.odx	07/09/2012 13:37	Arquivo ODX	312 KB
EV_ECM20TDI03L9C6019AG_A01104.odx	07/09/2012 13:37	Arquivo ODX	311 KB
EV_ECM20TDI03L9C6019AH_A011C4.cdx	07/09/2012 13:37	Arquivo ODX	313 KB
EV_ECM20TDI03L9C6019AJ_A01104.odx	07/09/2012 13:37	Arquivo ODX	311 KB
EV_ECM20TDI03L9C6019AK_A01104.odx	07/09/2012 13:37	Arquivo ODX	312 KB
EV_ECM20TDI03L9C6019L_A01202.odx	07/09/2012 13:37	Arquivo ODX	311 KB

I found a nice free odx reader: https://www.intrepidcs.com/products/free-tools/tiny-odx-viewer/

On the excel attached below you have the models and years of each package sent by VW.

As for disassembly, the best tool used today is the IDA Pro. Their version 5 is freeware. I don't know if it has good use, because you need match the microcontroller used for best disassembler results. I tried for a couple of settings and didn't give good results, but it is worth the try. https://www.hex-rays.com/products/ida/support/download_freeware.shtml

Regards,

Renan

On Fri, Sep 22, 2017 at 6:54 PM, Brian Chang-Yun Hsu <changyun.hsu@gmail.com> wrote:
Sure. I actually just followed the instructions here:

https://github.com/devttys0/binwalk

I am in the process of leaving a message there. The developer of binwalk is on there so hopefully he can answer it.

Brian Chang-Yun Hsu

On 22 September 2017 at 18:49, Jason from Collision Sciences <jbayley@collisionsciences.ca> wrote:
Renan,

If you were able to open the file, what years, models does the VW data apply to? I will ask them Monday if that data is available through ETI or just privately.

And Brian,

Can you tell Renan what steps you took to install the firmware analysis tools, so he can try...

J

On Sep 22, 2017 6:33 PM, "Brian Chang-Yun Hsu" <changyun.hsu@gmail.com> wrote:
Hi everyone,

I am actually stuck because for some reason I cannot install binwalk on my Linux virtual machine. This is supposed to be a simple one-line command, but it is not working as I expected. I will post to a forum to try to get some help, but before that I am afraid I cannot move further on the firmware stuff.

Again, if we can find software to open those ODX/PDX files, we probably do not need CANalyzer for the moment. I was going to play with CANtact to see its capabilities, but I still have not gotten my Ontario driver's license so I cannot pick up my car yet. Once I have it, I will go pick it up from Jason (and we might be able to all meet in person then).

Thanks!

Brian

On 22 September 2017 at 18:22, Jason from Collision Sciences <jbayley@collisionsciences.ca> wrote:
Renan,

Were you able to open the file? If so, can you please let Brian know which software.

Sounds like good news.

Brian has been making headway on the firware (Brian, please update Renan on the process and progess...)

Also, you both have had good inputs on which CANalayzer tool we could benefit from. Can you please both remind me and document your thoughts here, for a discussion on what we need.

J

On Sep 22, 2017 4:40 PM, "Renan Pedrosa" <renanjpedrosa@gmail.com> wrote:
Hi Jason and Brian,

This info is very good. If you can get packages like these from other manufacturers you will be able to generate the full reports like Bosch does. If you want to build your own reports, that's what you would be looking for, several ODX files for the Airbag ECUs. It would take a good amount of time to extract the information, but it is all there (like what is the PID for acceleration, brake, history, what is the conversion, etc...). The only information that I don't know if it is there is the seed key. For this first moment, the ODX can help a lot understand the communication between the Software, CDR and Car.

Regards,

Renan

On Thu, Sep 21, 2017 at 2:43 PM, Jason from Collision Sciences <jbayley@collisionsciences.ca>
wrote:
Hi Brian/Renan,

Meet each other and help each other! You both habe similar skill sets. We can all meet in person
together soon.

VW seems to be the most helpful so far... they uploaded a sample odx for us, see below.

Jason

Hello
This is a secured FTP so you'll need a secured FTP client to access the site.
The sample data will be removed from the site on Monday.

IP: **8.27.178.128**

User ID: Generic   Password: sw5guN+bEw

Let me know if you need anything else.
---------- Forwarded message ----------
From: "Mondary Janice (AA-AS/TSS6-NA)" <janice.mondary@us.bosch.com>
Date: Sep 21, 2017 2:38 PM
Subject: RE: Crash Data Info (for Invent with Bosch)
To: "Jason from Collision Sciences" <jbayley@collisionsciences.ca>
Cc:

Hello
This is a secured FTP so you'll need a secured FTP client to access the site.
The sample data will be removed from the site on Monday.

IP: **8.27.178.128**

User ID: Generic   Password: sw5guN+bEw

Let me know if you need anything else.

Janice Mondary

Tel. 586-578-7448 | Mobile 586-246-4016

From: Jason from Collision Sciences [mailto:jbayley@collisionsciences.ca]
Sent: Thursday, September 21, 2017 2:28 PM
To: Mondary Janice (AA-AS/TSS6-NA) <janice.mondary@us.bosch.com>
Subject: RE: Crash Data Info (for Invent with Bosch)

Hi Janice,

I believe my engineers have an ODX reader, or do we need the version from this company specifically for your data? I have reached out to them.

If you setup ftp in meantime, that would be great.

Jason

On Sep 21, 2017 10:33 AM, "Mondary Janice (AA-AS/TSS6-NA)"
<janice.mondary@us.bosch.com> wrote:

Hello

I can put some sample data on a secured FTP once you have the ODX reader.

You'll need a ODX reader to view the data. If you're an ETI member (https://www.etools.org) , you can get a license free.

Additional licenses must be purchased from RA directly. Use the following link or phone:

http://www.rac.de/en/support/contact

Help desk +49 7251 386215 (German company so German office hours apply)

Janice Mondary

Tel. 586-578-7448 | Mobile 586-246-4016

From: Jason from Collision Sciences [mailto:jbayley@collisionsciences.ca]
Sent: Wednesday, September 20, 2017 12:30 PM
To: Mondary Janice (AA-AS/TSS6-NA) <janice.mondary@us.bosch.com>
Subject: Re: Crash Data Info (for Invent with Bosch)

Hi Janice,

Hope you are well. Just following up on the diagnostic info we requested.

When we talked on the phone, you were going to send by mail some information.

I offered to provide a code snapshot to ensure we receive the parameters needed for the vehicle process. How's it going with that?

If you were unable to confirm you are sending everything we need, maybe you could just send what you have and we can try and confirm on our end?

Jason

.

On Thu, Sep 7, 2017 at 2:56 PM, Jason from Collision Sciences <jbayley@collisionsciences.ca> wrote:

Hi Janice,

As I mentioned, this project was inspired by the "Invent with Bosch" program. https://www.boschdiagnostics.com/cdr/invent-bosch

I have not yet submitted the project, as I don't feel it is yet ready. We have a full working solution, but for limited vehicles.

With advanced diagnostic VW data (CANID and DIAGNOSTIC map) as per the the data shown in the attached screenshot, I would be able to create a VW-specific database and provide a good prototype, where the described solution could be tested on a range of vehicles.

If you would like to send the data to me, any digital formal is good. You can mail to:

Jason Bayley
192-1055 Shawnmarr Road
Mississauga, ON
L5H3V2
Canada

Thanks,

Jason

Jason Bayley
Director, CollisionSciences.ca

M: 1.905.599.9899
2680 Matheson Blvd E. Suite 102
Mississauga, ON Canada L4W 0A5

--

Jason

Jason Bayley
Director, CollisionSciences.ca
M: 1.905.599.9899
2680 Matheson Blvd E. Suite 102
Mississauga, ON Canada L4W 0A5

--

Renan Pedrosa

--

Renan Pedrosa

--

Renan Pedrosa

--

Jason

Jason Bayley
Director, CollisionSciences.ca
M: 1.905.599.9899
2680 Matheson Blvd E. Suite 102
Mississauga, ON Canada L4W 0A5

--

Jason

Jason Bayley
Director, CollisionSciences.ca
M: 1.905.599.9899
2680 Matheson Blvd E. Suite 102
Mississauga, ON Canada L4W 0A5

Cool. Thanks!

On Jan 26, 2018 16:29, "Jason Bayley" <jasonmbayley@gmail.com> wrote:
---------- Forwarded message ----------
From: <BoschLicenses@us.bosch.com>
Date: Jan 26, 2018 11:57 AM
Subject: CDR Software Subscription Activation Certificate for version 17.6
To: <jasonmbayley@gmail.com>
Cc: <cdgvendor@crashdatagroup.com>

Dear Jason Bayley,

Thank you for your recent purchase of Bosch CDR software subscription(s).

Our records indicate you have purchased the following:
(1) 1 Year CDR Software Subscription(s)

All software subscriptions entitle you to CDR software updates that become available for your indicated subscription period ending on 1/26/2019 10:57:53 AM.

Attached is an Activation Certificate you will need to activate your new CDR software version 17.6 on each computer you are entitled to run the software on. The CDR program explains this in the End User License Agreement which is displayed to you during the activation and you must agree to it before the software can be activated.

For as long as your subscription is current, you will receive a similar email every time a new version of CDR software becomes available. Please save this email and the attached Activation Certificate for future reference and incase you want to transfer the licensed software from one computer to another.

To download, install and activate your CDR software, follow the steps below:

1.   Save the Activation Certificate attached to this email to a folder or to your desktop on your computer. You will need this once you have installed the CDR software.

2.   Click http://www.boschdiagnostics.com/software/Pages/CDR_software.aspx or, go to www.boschdiagnostics.com and click on the "Crash Data Retrieval" link on the left hand pane on the webpage.

3.   Unzip the downloaded file and run the setup.exe program to install the software and follow the installation instructions on screen.

4.   Double-click the file you just downloaded and follow the installation instructions on screen.

5.   Once the CDR software is installed, start the new CDR software program and click the "Activate" button displayed on the screen after the program is launched.

6.   Follow the instructions on the screen and, when prompted, open the Activation Certificate to activate your CDR program.

Hrg. EX.

52

If you need more detailed instructions on installing your CDR software, click on the link below to download an instruction manual.

https://www.boschdiagnostics.com/cdr/sites/cdr/files/CDR_software_Installation_Manual.pdf

If you encounter any problems installing and activating your software, please contact Bosch Diagnostics Technical Support at 1 (855) 267-2483 (toll-free US & Canada) or +1(805) 966-2000 ext. 4, option 1 (international callers).

Sincerely,

The CDR Product Team of Bosch Automotive Service Solutions

G2058 (Collision Sciences)

Renan,

I tested the code but I could not get the CDR to go past 2A 01 40 or so, possibly due to a timeout.
I think the CDR has a timeout set for 30 seconds, during which one scan pass must complete. The Python program for some reason is not fast enough so the CDR kept timing out.
I am using CDR 17.9.1, FYI. I am also attaching a trace that I got from the Python program.

Thanks!

Brian Chang-Yun Hsu

On Thu, Nov 22, 2018 at 9:35 AM Renan Pedrosa <renanjpedrosa@gmail.com> wrote:
Hey Guys,

Got the CDR Replay for VPW working. So far it reads the json file, finds the correct requests and responses, replies correctly to the CDR and the CDR continues the process. While doing that, it also sends a keep alive message to signal to the CDR that it os still there.

The used files are cdr-replay.py and Obd_dongle.py. If there are any problems, please let me now.

https://drive.google.com/folderview?id=1ciExerSFWCfDqqlb6aNPc8wDX6vAfBdL

Thanks,

Renan

Hrg. EX.
54

From:	Brian Chang-Yun Hsu [changyun.hsu@gmail.com]
on behalf of	Brian Chang-Yun Hsu <changyun.hsu@gmail.com> [changyun.hsu@gmail.com]
Sent:	2/21/2018 2:24:18 PM
To:	Chris Hsu [endor1984@gmail.com]
Subject:	Bosch CDR Report
Attachments:	2T3RFREVXHW656227_ACM_2017ToyotaRAV4.PDF

This is one service that we would like to provide eventually. Users or claim investigators can scan cars and get some preliminary results. If they need a formal report (to be used in court, for example), they can pay us and we will generate the actual PDF report for them.

The attached PDF is the report for your car. Raw data from the scan are stored in the database on the server, so I can use those data and "replay" back to the Bosch tool and make it generate a report.

Michelle's car should contain actual crash data so there will be graphs and tables...etc. in the actual PDF report.

Brian Chang-Yun Hsu

Hrg. EX.
61

IMPORTANT NOTICE: Robert Bosch LLC and the manufacturers whose vehicles are accessible using the CDR System urge end users to use the latest production release of the Crash Data Retrieval system software when viewing, printing or exporting any retrieved data from within the CDR program. Using the latest version of the CDR software is the best way to ensure that retrieved data has been translated using the most current information provided by the manufacturers of the vehicles supported by this product.

CDR File Information

User Entered VIN/Frame Number	2T3RFREVXHW656227
User	
Case Number	
EDR Data Imaging Date	02/21/2018
Crash Date	
Filename	2T3RFREVXHW656227_ACM_2017TOYOTARAV4.CDRX
Saved on	Wednesday, February 21 2018 at 09:08:54
Imaged with CDR version	Crash Data Retrieval Tool 17.6.1
Imaged with Software Licensed to (Company Name)	Collision Sciences
Reported with CDR version	Crash Data Retrieval Tool 17.6.1
Reported with Software Licensed to (Company Name)	Collision Sciences
EDR Device Type	Airbag Control Module
Event(s) recovered	None

Comments

No comments entered.

Data Limitations
CDR Record Information:

- Due to limitations of the data recorded by the airbag ECU, such as the resolution, data range, sampling interval, time period of the recording, and the items recorded, the information provided by this data may not be sufficient to capture the entire crash.
- Pre-Crash data is recorded in discrete intervals. Due to different refresh rates within the vehicle's electronics, the data recorded may not be synchronous to each other.
- Airbag ECU data should be used in conjunction with other physical evidence obtained from the vehicle and the surrounding circumstances.
- If any of the front passenger seat airbags, side airbags, or Curtain Shield Airbags have deployed, data will not be overwritten or deleted by the airbag ECU following that event. If none of the airbags have deployed, the data of that event may be overwritten by a following event even if other airbags (pretensioner, rear seat airbag, etc.) have deployed.
- If power supply to the airbag ECU is lost during an event, all or part of the data may not be recorded.
- "Diagnostic Trouble Codes" are information about faults when a recording trigger is established. Various diagnostic trouble codes could be set and recorded due to component or system damage during an accident.
- The airbag ECU records only diagnostic information related to the airbag system. It does not record diagnostic information related to other vehicle systems.
- The TaSCAN, Global Tech Stream, or Intelligent Tester II devices (or any other Toyota genuine diagnostic tool) can be used to obtain detailed information on the diagnostic trouble codes from the airbag system, as well as diagnostic information from other systems. However, in some cases, the diagnostic trouble codes of the airbag system recorded by the airbag ECU when the event occurred may not match the diagnostic trouble codes read out when the diagnostic tool is used.

General Information:

- The data recording specifications of Toyota's airbag ECUs are divided into the following categories. The specifications for 12EDR or later are designed to be compatible with NHTSA's 49CFR Part 563 rule.
  - 00EDR / 02EDR / 04EDR / 06EDR / 10EDR / 12EDR / 13EDR / 15EDR / 17EDR
- The airbag ECU records data for all or some of the following accident types: frontal crash, rear crash , side crash, and rollover events. Depending on the installed airbag ECU, data for side crash and/or rollover events may not be recorded.
- This airbag ECU records record pre-crash data and post-crash data.
  - If a single event occurs independently, the data for that event is recorded on a one-to-one basis.
  - If multiple events occur successively (within a period of approximately 500ms), the establishment of the recording trigger for the first event is defined as the "pre-crash recording trigger". Pre-crash data for the first event and post-crash data for each successive event is then recorded.
- The airbag ECU has two recording pages (memory maps) to store pre-crash data. Additionally, to store post-crash data, the airbag ECU has two recording pages for each accident type: two pages for frontal and rear crash, two pages for a side crash, and two pages for rollover event.
- The data recorded by the airbag ECU includes correlating information between each previously occurring event (i.e., information that clarifies the collision event sequence. This correlation information consists of the following items.
  - Time from Previous Pre-Crash TRG
  - Linked Pre-Crash Page
  - Time from Pre-Crash TRG
  - TRG Count
  - Previous Crash Type

INVOICE

MAILING INFO	**Brian Chang-Yun Hsu**	**BILL** TO	**Collision Sciences Inc.**
	26 Beechwood Avenue		192-1055 Shawnmarr Road
	Toronto, ON M6N 4T1		Mississauga, ON L5H 3V2
	Phone: (647) 898-4560		Phone: (905) 599-9899
PAYPAL ID	changyun.hsu@gmail.com		

DATE	DESCRIPTION	HOURS	RATE	AMOUNT
2018/04/17	PayPal Transction Fee	N/A	$72.63	$72.63
2018/04/16		8.5	$30.00	S255.00
2018/04/17		8.5	$30.00	S255.00
2018/04/18		7.5	$30.00	S225.00
2018/04/19		7.0	$30.00	S210.00
2018/04/20		6.0	$30.00	S180.00
2018/04/21		3.0	$30.00	$90.00
2018/04/22		6.5	$30.00	S195.00
2018/04/23	See Attachment for Details	9.0	$30.00	S270.00
2018/04/24		9.0	$30.00	S270.00
2018/04/25		8.0	$30.00	S240.00
2018/04/26		11.0	$30.00	S330.00
2018/04/27		9.0	$30.00	S270.00
2018/04/28		3.0	$30.00	$90.00
2018/04/29		4.0	$30.00	S120.00
2018/04/30		8.0	$30.00	S240.00
		108.0	SUBTOTAL	$3,312.63
			TAX RATE	0.000%
			TAX $	-
			DISCOUNT $	-
			TOTAL $	3,312.63

OTHER COMMENTS

1. Tax for credit card to be agreed with client.

2. Please include the invoice number on your check

3. To pay via PayPal, send the total to changyun.hsu@gmail.com

Make all checks payable to:
Brian Chang-Yun Hsu

Thank You For Your Business!

Hrg. EX.
69

Brian Chang-Yun Hsu Work Log
April 16, 2018 to April 30, 2018

$72.63 of PayPal fee from previous payment

April 16 [9:00am - 1:00pm; 1:30pm - 5:30pm; 7:00pm - 7:30pm] [8.5 Hours]

Investigated CDR replay for Toyota Cable 617. This cable uses K-Line, but is different from the other K-Line data that can be replayed using CAN. The CDR stopped asking PIDs after 0x0F, but it was supposed to ask more, so replaying back to the CDR is an issue right now.

Made a replay file for Cable 617 so that I could test the app and debug the crash AI.

Found an error in the calculation of longitudinal delta-V for Cable 617, 00EDR and 02EDR. Fixed the error in vehicle.py.

Got Toyota Cable 617 crash AI working. In addition to the current data structure, the Python code makes another JSON structure containing crash data grouped by events. For Cable 617, the pre-crash data is right after longitudinal delta-v values, so the pre-crash data and actual crash data can be matched easily and accurately, unlike the newer Toyota vehicles. Rollover events do not have pre-crash data, so they will just have the roll angle and the roll lateral acceleration in m/s².

EDIT: Cable 617 is a special kind of protocol that we cannot do yet, so the crash AI is not useful at all at this point.

Checked Fedir's work against the previous copy of the repository.

      Extension -> UIFont+CrashScan.swift
      Extension -> Array+CrashScan.swift
      Managers -> StorageManager.swift
      Managers -> NotificationsManager.swift
      OBD -> BTLESerialTransporter.swift
      Scenes -> Main -> Scan -> ScanTabPresenter.swift
      Scenes -> Main -> Scan -> ScanTabViewController.swift
      Scenes -> Main -> Result -> ResultTabPresenter.swift
      Scenes -> Main -> Result -> ResultTabViewController.swift
      Scenes -> Main -> Result -> Details -> ResultTabStoryboard.storyboard

      Added LTSupportAutomotive library

Checked the trace from Chad's GMC Sierra. The received data are no longer mismatched, possibly due to the 'ATAT 0' command. However, for the actual crash data PIDs (0x31, 0x32, and 0x33) where around 1200 bytes of data are expected, the responses were cut off. None of the three PIDs managed to capture the entire response. I will most likely have to test on a real car for this issue.

Tested Devendra's code and noticed a few issues. Most obvious one was that the 'Scan Vehicle' button did not become re-enabled if the task threw an Exception. He fixed that and I tested again to make sure it worked. However, there were still some minor issues with the status display text (which I know how to fix).

April 17 [9:00am - 2:00pm; 2:30pm - 5:30pm; 6:00pm - 6:30pm] [8.5 Hours]

Made an Installrapp package for Devendra's Bluetooth version. Eventually the two branches will be merged.

Drove to Finelines Auto to try the Honda K-Line using methods suggested by the OBDLink team. I tried their suggestion as well as a few other things, but the module would not respond to any requests.

Met up with Chad and tested on his GMC Sierra Denali. The version on my laptop worked perfectly and got all of the bytes from the three longest PIDs (the ones containing crash data). His Android phone would not connect to Bluetooth at all, so we will see if Devendra's version helps with that.

Downloaded the Dash app and tested various cases for the Bluetooth.

1. Ran the Dash app to set up. Paired with OBDLink LX.
2. Closed the Dash app, but left it running in the background (the Dash logo was still showing in the status bar)
3. Launched CrashScan. My last used device was OBDLink MX, so I chose OBDLink LX in the dropdown menu.
4. Choosing the LX made the Dash app close itself (the Dash logo was not in the status bar anymore). Our app could get the VIN.
5. Relaunch the Dash app, with CrashScan running in the background.
6. Dash app was able to regain control of the LX and got vehicle information.
7. I closed the Dash app (Dash logo in status bar) and try to get the VIN again from our app. Our app hangs on "Resetting OBD Connection"
8. I closed the Dash app completely (no logo in status bar) and went back to the app. Same issue.
9. If I 'Force Stop' and relaunch the app, the app works fine.
10. Now that the default device in CrashScan is the LX, I repeated the test.
11. When I launched our app, the Dash app was closed (no logo on the status bar), but our app showed "Not Connected".
12. About 10 seconds later, the Dash app relaunched itself and appeared to have connected to the LX again.
13. I then tapped on the OBDLink LX text in our app and made it reconnect. The app reconnected with the LX and was able to get the VIN. However, as soon as I pressed 'Scan Vehicle', the app stopped at "Resetting OBD connection…"
14. Then after a short while, the Dash app regained control of the dongle.

When stuck at "Resetting OBD connection…", this is the error printed out in Android Studio:

```
D/BluetoothService: Queueing jobs for connection configuration..
W/System.err: java.io.IOException: Broken pipe
W/System.err:    at android.net.LocalSocketImpl.writeba_native(Native Method)
W/System.err:    at android.net.LocalSocketImpl.-wrap3(Unknown Source:0)
W/System.err:    at android.net.LocalSocketImpl$SocketOutputStream.write(LocalSocketImpl.java:148)
W/System.err:    at android.bluetooth.BluetoothSocket.write(BluetoothSocket.java:582)
W/System.err:    at android.bluetooth.BluetoothOutputStream.write(BluetoothOutputStream.java:85)
W/System.err:    at java.io.OutputStream.write(OutputStream.java:75)
W/System.err:    at ca.collisionsciences.obd.commands.ObdCommand.sendCommand(ObdCommand.java:108)
W/System.err:    at ca.collisionsciences.obd.commands.ObdCommand.run(ObdCommand.java:88)
```

```
W/System.err:    at ca.collisionsciences.app.bt.BluetoothService$GetVinTask.doInBackground(BluetoothService.java:1779)
W/System.err:    at ca.collisionsciences.app.bt.BluetoothService$GetVinTask.doInBackground(BluetoothService.java:1766)
W/System.err:    at android.os.AsyncTask$2.call(AsyncTask.java:333)
W/System.err:    at java.util.concurrent.FutureTask.run(FutureTask.java:266)
W/System.err:    at android.os.AsyncTask$SerialExecutor$1.run(AsyncTask.java:245)
W/System.err:    at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1162)
W/System.err:    at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:636)
W/System.err:    at java.lang.Thread.run(Thread.java:764)
E/DTCERR: Broken pipe
```

Emailed Brian at OBD Solutions for suggestions on Honda K-Line and SWCAN flow controls.

Replayed the data from Chad's truck to make sure the received data was complete for the CDR.

Made ReplayFileMaker.py and EEPROM.py into the same file, so that AutoIT only needs to run one file.

Went over the updated Bluetooth code with Devendra through TeamViewer.

April 18 [9:00am - 11:00am; 12:00pm - 5:30pm] [7.5 Hours]

Emailed back Brian Lee at OBD Solutions regarding Honda K-Line and SWCAN flow control issues.

Started testing updated code on the master branch. Observations:

1. If dongle is in idle/sleep mode, app cannot wake up the dongle and connect to it. Connection status text shows "Connecting..." then "Not Connected". This does not happen very often as regular users would move from car to car.
   There are two sleep timers for the LX. UART sleep occurs after 600 seconds (10 minutes) of inactivity. VL sleep occurs after 10800 seconds (3 hours) of voltage level being less than 13 V. In the current setup I have, voltage level is always less than 13 V (around 12.2 V).
2. Upon launching the app, dongle gets connected automatically. However, the progress bar remains visible and spinning, and the status text reads "Beginning to connect to OBD device..."
3. If the app failed to get the VIN (in this case I forgot to run Busmaster), the status text would stay at "Getting VIN..." with the progress bar spinning. The 'Query VIN' button was enabled so I could just try again.
4. If VIN retrieval is successful, the status text shows "Complete!" with the progress bar spinning.
5. Drive Clean process has display text "Connecting to server...", which is what we want.
6. CrashScan process does not have percentage anymore.
7. The close button on the scan result popup dialog is not very sensitive. The first time I scanned I had to tap 3 times to close the dialog. The second time I had to tap 5 times to close it. There were no error messages in Android Studio for this issue.
8. Scan results do show up immediately afterwards. However, the sample entries are also re-generated so they always remain at the top.
9. After a scan is completed and the dialog is closed, going back to the 'Scan' tab shows the progress bar spinning with status text "Complete!", similar to the VIN task.
10. With all dongles connected at the beginning, I can switch between dongles and get the VIN easily. I then unplugged the LX, plugged it back in, waited for it to fully boot up ('Power' LED becomes steady and not blinking), and tried to connect to it. The first time I clicked on 'Query

VIN', Android Studio gave an error of "Bluetooth socket closed, read return: -1". I clicked the button again and got a "Broken pipe" error. At this point, unplugging and plugging in the dongle does not resolve the issue. Closing the app (with it running in the foreground service) and opening it again does not resolve the issue. Only by stopping the app completely (currently requires 'Force Stop') can the dongle work again.

11. When app is connected to the LX and running the foreground service, Realterm was unable to take over the connection.

Got the VIN ignition error message to pop up if VIN retrieval is unsuccessful. Changed the foreground service icon to the car icon.

Made ATAT 0 available for all makes in CrashScan.

Fixed several issues above.

April 19 [8:30am - 12:30pm; 1:00pm - 4:00pm] [7 Hours]

The 2010 Acura MDX had 'Call for more details' in the popup dialog, but it really should be 'Unsupported'. Checked vehicle.py and corrected the error. Checked other processes as well to make sure unsupported vehicles bypass CrashScan and go straight to Drive Clean.

Made a MINI lookup table to check if a particular MINI is supported for crash data. Due to the naming complication in MINI vehicles, the cable lookup code is slightly different. Tested to make sure the program would be accurate in determining support. Tested a couple BMW VIN as well since BMW and MINI are sharing the same process in vehicle.py.

Dialog was not showing if CrashScan is unsupported. Made changes and tested the following combinations.

Drive Clean	CrashScan	Result
Successful	Successful	OK
Successful	Failed	OK
Failed	Successful	OK
Failed	Failed	OK

Resending buffer full commands should be limited to 3 times because if someone accidentally uses the Tonwon or Carista for CrashScan, the app will hang indefinitely. When buffer full error occurs 3 times for a given command, the app will display a popup dialog warning users that information received could be inaccurate and that they should use dongles with larger buffers.

Incorrect VIN length or other VIN errors did not show alert popup dialog. Fixed that in Android Studio.

April 20 [9:00am - 2:00pm; 3:00pm - 4:00pm] [6 Hours]

Decoded pre-crash data for Audi. This is not yet implemented in vehicle.py.

Went to Stark Auto Sales for field testing. Got a 2006 Toyota Matrix and a 2013 Ford Escape. The app did not report the correct longitudinal delta-v (it was not the max value) for the Matrix, and the popup did not show up at all for the Ford Escape.

Checked the error log for the scans today. The Ford longitudinal delta-v values were completely wrong (~300,000 km/hr), and the Toyota Celica did not respond to our request for part number and thus the process failed. In fact, I was unable to even get the VIN and had to manually enter it in the app. The car might not have been able to communicate at all.

April 21 [9:00am - 12:00pm] [3 Hours]

The 2013 Ford Escape scanned yesterday and the 2014 Ford Escape share the same part number, but the crash data locations are different. Compared to other vehicles that use cable 783, the crash data locations are all different. I marked them as 'M' for manual processing via CDR replay and just not report anything in app for the time being.

Fixed the Matrix resolution issue. We actually have seen another part number that did this, so I added this part number to our list of exceptions. We will just have to add to this list as we encounter more.

Added a setting in the app code to force the screen to stay on. Otherwise, the app will crash when trying to display the final popup dialog.

April 22 [9:00am - 12:00pm; 1:00pm - 4:30pm] [6.5 Hours]

Tested Devendra's code to make sure switching between cars does not cause Bluetooth issues. The first time I tried the code, the app got stuck at "Resetting OBD Connection...", but all subsequent trials went fine.

Implemented Audi pre-crash data in vehicle.py.

Discussed with Renan about the VIN retrieval dongle issue. Having a longer time delay does not help with the issue. I tried to step through the program slowly, but still got the same result.

Went back to Drive Clean web server tests. Completed two more tests, but we still have three tests that do not work.

Decoded and implemented pre-crash data for two types of GM cable 454 modules.

April 23 [9:00am - 5:00pm; 8:00pm - 9:00pm] [9 Hours]

Field day at Stark Auto Sales.

Reviewed the MS-CAN for Ford and BMW traces.

April 24 [10:00am - 5:00pm; 6:00pm - 7:00pm; 8:00pm - 9:00pm] [9 Hours]

Field day at Stark Auto Sales.

Fixed 2012 and 2013 Mazda lookup sheet.

Fixed Ford MS-CAN lookup table by adding MS-CAN vehicles.

April 25 [9:00am - 1:00pm; 3:00pm - 7:00pm] [8 Hours]

Fixed Lincoln MS-CAN lookup table by adding MS-CAN vehicles.

Added 2008 Saturn Vue as an exception in vehicle.py that uses a different resolution for delta-v.

Make Ford and Lincoln MS-CAN a trial-and-error step in app. If regular CAN leads to 'NO DATA' from the module, the app will try to switch to MS-CAN. However, if the dongle does not support MS-CAN, the app will just display a message and abort.

I noticed that even if a user does not have email and password, the app will still run and even produce the popup dialog. The database will not have an email address for these scans.

Tried to replay the 2011 Toyota Corolla, which uses K-Line, but the ReplayFileMaker got an error. After some analysis I found the error and corrected it. The replay file then was made and replayed successfully. I then noticed that the resolution was a little bit off. I recalculated the resolution and updated vehicle.py.

Replayed a 2015 Ford Focus, but the CDR asked for an additional PID at the very end (22 F1 25). I added that PID to our request list. Since there could be more PIDs after the F1 25, we need to find similar cars and take a trace to get all PIDs. This vehicle uses cable 783.

Replayed a 2015 Nissan Sentra without any issues.

Replayed a 2012 Ford Escape without any issues. This vehicle uses cable 384.

April 26 [9:00am - 5:00pm; 6:00pm - 9:00pm] [11 Hours]

Checked the GMC Canyon data trace and found the error with GM VPW process in the app, but we would still have to try on one real vehicle to verify.

Field day at Stark Auto Sales.

Fixed BMW crash AI.

Fixed pending response issue in Toyota.

Got started with the 2013 Corolla Matrix resolution issue. The longitudinal delta-v is not interpreted the same way as other EDR 12 modules. This module uses cable 613, which is technically a K-Line cable. I still need to figure out how to decode the delta-v from this module.

April 27 [8:00am - 5:00pm] [9 Hours]

Made a minor adjustment to the pending response fix implemented yesterday. This makes the raw data removal more robust to prevent unintentional deletion of data.

2014 Toyota Matrix had a side impact, but the app did not report anything. This is due to pending response not completely removed from the raw data (fixed yesterday). Replayed this car's data and confirmed that the app would report the correct side impact delta-v.

Implemented newly discovered FCA CAN ID in vehicle.py. The upgraded FCA process now checks three sets of CAN IDs, and then based on the responses it determines which set of PIDs to use. If all three CAN IDs cannot get any response from a car, the process ends.

Checked the 2015 Ford Focus trace. It looks like the CDR only asks F1 25 at the end of the process and no other PIDs, so the current PID list is okay.

Field day at Stark Auto Sales.

April 28 [10:00am - 1:00pm] [3 Hours]

Fixed the 2012 Dodge Caravan resolution issue. App showed -18 km/hr, but report said -14 km/hr. This is due to the part being supplied by Continental Corporation, which uses mph on newer FCA vehicles. I added the part number from this vehicle to our "mph" list in vehicle.py.

2011 Lexus IS250 should have double the resolution for longitudinal delta-v. Added its part number as an exception in vehicle.py

2008 Chevy Impala report shows -4.99 mph but app said -7.00 km/hr for delta-v. This is the same issue as the 2008 Saturn Vue. Added this vehicle to the exception list as well.

Fixed ReplayFileMaker for the 2011 Lexus IS250. This module uses different requests for EEPROM, so I added a new list of indices for this module.

April 29 [9:30am - 11:30am; 8:00pm - 10:00pm] [4 Hours]

Improved ReplayFileMaker for additional Toyota EEPROM part (part type 14).

Fixed the blue HP laptop by removing HHD virtual serial port software and deleting a registry value.

Finally decoded and solved the 2013 Corolla Matrix issue. These modules use 14-bit two's complement instead of regular 8-bit or 16-bit. I think this applies to all modules that use cable 613 but have 12EDR.

April 30 [9:00am - 4:30pm; 8:00pm - 8:30pm] [8 Hours]

Fixed ReplayFileMaker for BMW to deal with different CAN ID used in the CrashScan process.

Field day at Stark Auto Sales.

Tested using the DLC connector and confirmed that we can test MS-CAN using Busmaster. We can run CAN commands through the DLC adapter for PCAN, and the MX can communicate with Busmaster using MS-CAN (STP 53) with a baud rate of 500,000 bps. Even in STP 53 mode, the dongle still prints out BUFFERFULL in text, so we would still be able to know when the buffer gets full regardless of whether a particular vehicle uses standard CAN or MS-CAN.

INVOICE

INVOICE # 20200407

MAILING INFO **Brian Chang-Yun Hsu**
324-36 Via Bagnato
Toronto, ON M6A 0B7

PHONE (647)-898-4560

EMAIL changyun.hsu@gmail.com

DATE 2020/04/07

BILL TO **Colllsion Sciences Inc.**
192-1055 Shawnmarr Road
Mississauga, ON L5H 3V2

PHONE (905)-599-9899

Date	Description	Hours	Rate	Amount
2020/03/16		10.0	$41.00	$410.00
2020/03/17		8.0	$41.00	$328.00
2020/03/18		8.0	$41.00	$328.00
2020/03/19		8.0	$41.00	$328.00
2020/03/20		7.0	$41.00	$287.00
2020/03/21		0.0	$41.00	$0.00
2020/03/22		1.0	$41.00	$41.00
2020/03/23		9.0	$41.00	$369.00
2020/03/24	See Attachment for Details	8.0	$41.00	$328.00
2020/03/25		8.0	$41.00	$328.00
2020/03/26		8.0	$41.00	$328.00
2020/03/27		8.0	$41.00	$328.00
2020/03/28		2.0	$41.00	$82.00
2020/03/29		0.0	$41.00	$0.00
2020/03/30		8.0	$41.00	$328.00
2020/03/31		10.0	$41.00	$410.00
		103.0	**SUBTOTAL**	**$4,223.00**
			TAX RATE	**13.00%**
			TAX	**$548.99**
			TOTAL	**$4,771.99**

**EXHIBIT
071**

Brian Chang-Yun Hsu Work Log
March 16, 2020 to March 31, 2020

March 16 [8:30am - 12:30pm; 1:00pm - 5:00pm; 8:00pm - 10:00pm] [10 Hours]

Went through the ISQ from Allstate and answered a couple of questions.

Checked and released 2 KPSA reports blocked due to high pre-crash speed and large variation in pre-crash speed.

GCNA scanned a 2020 Lexus ES 300h. The part number is not on the CDR's list, but the CDR was able to process it and generated a report (no crash data). Added the part number in the database. This module is 17EDR.

Replayed and released a 2004 Toyota Camry scanned by AmFam.

Experimented with adding the curl command for Armor Anywhere Agent installation directly in our .ebextensions folder, which contains commands that would be run every time the code is deployed to Elastic Beanstalk. The command actually worked and managed to install the agent. The Armor dashboard confirmed the installation and showed the correct IP for the EC2 instance.

Contacted Armor and got a response on how to remove the agent. Added the 2 commands for removing the agent, and the commands worked perfectly. The Armor dashboard did not show the agent anymore.

Replayed a 2014 Nissan Versa Note and released the report.

Planned out an update to the scan process regarding report fees and purchases:

After a scan, if a user is a subscribed type, the code would automatically create a row in the 'report_purchase' table with the correct fee and currency. In the event where processing fails, the code should not attempt to determine the report fee, even if the user is not subscribed.

Then, in the 'release_report' endpoint, it should call a function to calculate the report fee and potentially update the information if there already exists an entry in the 'report_purchase' for the particular VIN.

Management reports can then use the date in the 'report_purchase' table whenever a date is needed for billing purposes. This means a VIN would have a scan date and a purchase date. If subscribed, the two dates would be the same.

Modified 'report.py' and 'report.html' to display the LVI version of the injury probabilities if certain conditions are met.

Added disclaimer for recording below the threshold in the 'How To' section in Crash Data Records.

Deployed Armor Anywhere Agent to the production environment. Added code in the .config file to only install the agent if the directory does not exist and if the environment is the production one.

March 17 [8:00am - 2:00pm; 3:00pm - 5:00pm] [8 Hours]

The production environment went into the Red state. Checked the enhanced health page and found out that the HTML to PDF package installation failed. Googled the issue and found out that the developers recently moved the file location to Github and the old path was lacking a security certificate, which AWS did not like.

Updated the file host location URL and redeployed the code to restore functionality.

Noticed that the production environment would constantly transition from OK to Warning. Checked the enhanced health monitoring page and saw that RAM usage was above 90%. It was possible that the high RAM usage caused Elastic Beanstalk to create a new instance this morning, which failed during deployment because of the PDF package issue.

Deleted and re-created IAM credentials for Shekar. It seemed that with MFA enforced, the option to mandate password reset on 1st login cannot be selected, because only after MFA is added does the user have permission to do that.

Upgraded the hardware for the production environment from t2.micro to t2.small, which has 2 GB of RAM. The production environment seemed to settled at about 1.5 GB of RAM usage after the upgrade. Trend Micro actually recommends having 5 GB of RAM.

With Shekar having access to the IAM page again, asked him to added an MFA device, but AWS reported a duplicate name in the MFA. Used AWS CLI to delete the existing, unlinked MFA device under his profile. He was able to add the MFA afterwards.

Did a brief TeamViewer session with Shekar to go over some of the backend code. Emphasized on the deployment config files in ebextensions and API security in general.

Filled out part of the Packetlabs questionnaire.

Started experimenting with calling APIs with JWT in the header in the Android app. Separated out the appConfig and representedClients payloads from the login endpoint. Modified the Android app to get these two things by calling two separate APIs. Might need to create a small API to just send back a JWT, without organization ID or country code.

March 18 [8:30am - 1:30pm; 2:00pm - 5:00pm] [8 Hours]

Reviewed and released two OM reports.

Continued working on adding authentication to API calls in Android. Researched about access token and refresh token.

Reviewed and released two AmFam reports.

Updated fee determination logic to include HUB. HUB will now use the same logic as AmFam.

Modified the code for Interceptor and moved it to a new 'TokenInterceptor' class that can be shared by other files.

For token-based API calls, there should be two types of tokens: access token and refresh token.

1. When a user logs in, the server checks the username and password.
2. If password is correct, the server generates an access token (JWT) and a refresh token (UUID).
3. The server stores the refresh token in the database, in the user table.
4. The server returns the access token and the refresh token.
5. If the access token is expired, the app sends the refresh token along with the access token.
6. The server gets the user ID from the access token and checks that the refresh token matches the one stored in the database.
7. The server generates and returns a new set of access token and refresh token. The database is updated with the new refresh token.

This solution makes it possible for the front end to not store username or password, but it will not work if HUB shares credentials.

Updated 3rd party injury text in the summary section. Modified the 3rd party injury section to show all types of collision (frontal, rear, and side).

March 19 [8:30am - 1:30pm; 2:00pm - 5:00pm] [8 Hours]

Modified the logic in global MG report as well as the client MG report to make HUB information show up correctly.

In the client MG report, HUB will have all scans they did (AmFam and non-AmFam), but only non-AmFam scans will have a fee if applicable. The global MG report would show the sum of the fees from non-AmFam scans.

Created a new subsection header for 3rd party injury section to distinguish between Most Recent event and LVI.

Updated variable names for the 3rd party injury section now that most recent event and LVI version both have separate values for front, rear, and side. Previously, the most recent event would make a guess on the crash type.

Updated the summary text for LVI cases to refer to the tables in the actual injury section.

Continued working on token authentication for the Android app.

Discussed with Chris about SharedPreferences and the work-in-progress. The proposed TokenInterceptor should ideally be in one file and shared by all API calls. However, since we have the 'app' module and the 'scan' module separated, this becomes somewhat complicated. Chris will take a look on the weekend.

March 20 [8:30am - 12:30pm; 1:00pm - 4:00pm] [7 Hours]

Checked a Maserati VIN for Theuns. The vehicle is a 2016 Maserati Ghibli, which is only supported in the US. Our database has the vehicle so that it can be manually selected. The app and backend can handle Maserati and we even had a GCNA scan before.

Pulled server logs for 2 KPSA scans that failed both DriveClean and CrashScan. In both cases, the app attempted module detection twice but could not establish communication with the EDR. DriveClean also failed in both cases, being unable to determine the protocol. It was most likely a power / key position issue.

Created a test web page and added the test URL from AWS to test the certificate for AWS S3. The test web page, when deployed to the dev environment, was able to show the test image normally.

Went back to decode pre-crash gear shift for Ford cable 783. Added code for both the normal cable 783 modules and the ones that look like 384.

Refactored the pre-crash gear shift code by making a dedicated function to return the position code ('P', 'R', 'N', 'D'...etc.) so to reduce repetition in the processing code.

Added Subaru models in the database according to the CDR support table. Duplicate rows were created if there were multiple market codes. South Africa has its own rows for all of these Subaru models and were marked supported.

Wrote a brief Google Document for Spencer and Kevin regarding the upcoming changes for token-based API authentication.

Experimented with JWT to create claims report links with expiration times. Ran into some issues with the timezone because the code would think our timezone is UTC and the reports would always be marked as expired. Used an alternative function to get the time at UTC by specifying explicitly.

The current time returned by the server has microseconds, but the 'exp' field is rounded to seconds after being encoded by JWT. Used the math.floor function to round the current time for comparison.

Created a new icon and new web page for the report expired page.

March 22 [2:00pm - 3:00pm] [1 Hours]

Reviewed and released a 2013 Ford Escape report by AmFam Ethos.

Replayed a 2001 GMC Yukon XL by AmFam Ethos. The scan contained no crash data, but the generated report still said manual review required. Checked the database and confirmed that the model 'Yukon' was in the database, but not 'Yukon XL'. Added 'Yukon XL' for model years 2001 to 2005.

With the cable number update, the processing code gave out some random values because it did not recognize that all data bytes were 0xFFs, meaning an empty module. Added more checks in the GM cable 829 processing code to check for 0xFFs.

Released the report for the 2001 GMC Yukon XL.

March 23 [8:00am - 12:00pm; 1:00pm - 6:00pm] [9 Hours]

Replayed and checked a 2018 Mazda CX-3 scanned by OM. The VIN and the part number were rejected by the CDR, so a replay report could not be generated. Checked the raw hexadecimal data directly. The data looked strange, but that was what was stored in the module. Notified Theuns of the situation and he would get pictures before we release the report.

Checked a 2018 Toyota Hilux scanned by MiWay South Africa. The delta-V and pre-crash data were all accurate. However, a couple of items in the seat belt and airbag table were different from the CDR.

Checked the raw data and the bytes would correspond to what our interpretation was, but the CDR regarded them as 'SNA'. Could not figure out how the CDR determined that. Sent Theuns the Bosch report in case he needed to proceed with the claim.

Got a 2018 Ford Figo from Old Mutual. The database did not have a cable assigned for it so the report said 'manual review required'. Calculated the response size of this module to be 1548, which was never seen before. There was a module with response size 1562 (2015 Lincoln Navigator) that appeared pretty similar to the Figo. Used some of the decoding scheme for 1562 and got the Figo report to display some more data. Notified Theuns of the update.

Checked a 2018 Ford Ranger for Theuns. The raw data confirmed that the module had no crash data as all bytes were 0xFFs.

Pulled the code that Chris worked on yesterday, utilizing dagger in Android.

Went through the Android code and resolved some bugs. JWT actually has a built-in check for expiration so I did not actually need to write extra code for that. However, for the same reason, the 'get_tokens' endpoint, which expects an expired token, would not work.

Added options to disable checks for expiration in endpoint 'get_tokens'.

Worked with Chris and hand off what I had so far so that he could test on his end. Modified a few endpoints in 'api.py' for the dev environment for testing the Android app.

March 24 [8:00am - 12:00pm; 1:00pm - 5:00pm] [8 Hours]

Decoded seat belt status for GM cable 004. Released the 2004 Saturn Vue report from a scan from yesterday.

Replayed a 2017 Hyundai Tucson for Theuns to confirm the direction of steering wheel angles.

Pulled the latest code from Chris' branch and tested the app with a full scan, since he did not have a vehicle simulation tool available. The access token was set to expire after 10 seconds intentionally, forcing the app to request a new one multiple times during a scan. The API calls worked perfectly and automatic token handling worked as expected.

Researched API versioning for our system. These endpoints will require a token and the Android app will start to call these instead. These could be used for our API penetration test.

Implemented a new path for version v1 of the APIs.

Replayed and reviewed 2 AmFam GM VPW reports. Both contained no crash data as confirmed by the CDR.

Updated the endpoint routes in Android to use the 'v1' endpoints, which require access tokens.

Merged branch 'dagger' back into 'develop-brian' for Android.

Went back to report link expiration and new report link format with JWT.

March 25 [8:30am - 12:30pm; 1:30pm - 5:00pm] [8 Hours]

Replayed and reviewed a 2014 Toyota Corolla Quest scanned by KPNA as it triggered an internal alert of large variation in pre-crash speed. CDR confirmed the data accuracy.

Theuns asked about a 2018 Ford Fiesta scanned by KPSA on March 12. This was one where CDR's interpretation appeared weird but our interpretation, based on a previously seen 2018 Ford EcoSport, was very reasonable. Theuns wanted to know whether cruise control was activated at the time. This report is billable for KPSA.

The report had a column for 'Speed Control Status', which I thought could actually be the cruise control status. However, after cross-checking available Ford cable 823 reports, that did not seem to be the case. There was a 2018 Ford Fusion where the driver overrode the cruise control by stepping on the accelerator pedal, meaning cruise control was on before the crash. However, the entire column of 'Speed Control Status' was all 'Off', so that did not appear to be related to cruise control at all.

Modified the old @requires_auth decorator as it did not actually check login credentials. Updated the 'authenticate' function to take a username and a password to verify credentials. This is shared by both the 'login' endpoint as well as the decorator itself.

Experimented with eliminating the 'retrieve_report' endpoint altogether. It allowed fast report retrieval, but clients are only given the 'create_report' link to ensure they always get the most up-to-date interpretation. In practice, only two endpoints would be needed, create and process.

Finished 'create_report' and 'process_report' on the v1 path, for both claims reports with JWT as ID and management reports with current format (API key plus optional date ranges).

Modified 'preview_report' to use the new JWT token format. Separated the token expiration code out into a new function so that it can be shared by endpoints.

Updated the code for email template processing code to include new text for report link renewal.

Added a dedicated row for OM in the email rule table, for report link renewal. Unlike the release report rule, this does not send emails to users / investigators. Only the CC and CS admin get emails.

Modified the 'process_email' code to send out appropriate messages and links for report link renewal emails.

March 26 [8:00am - 12:00pm; 2:00pm - 6:00pm] [8 Hours]

Checked a 2017 Mazda 3 scanned by MiWay South Africa. Decided to change the airbag deployment information from 'Unavailable' to 'No Deployment' if the raw byte is 0xFF, despite the CDR showing it as SNA.

Checked the raw data for the 2017 Lamborghini scanned by Theuns. The module rejected 2 of the 6 event requests. Theuns used the CDR and got some data. Like the rest of overseas Audi modules, the EDR only contained some date and time information, with no actual delta-V or pre-crash data.

Checked the raw data from other Audi scans and identified why the issue occurred.

Checked the raw data for a 2014 Ford Ranger for Theuns. The pre-crash data bytes were all 0s so there was no speed to report.

Checked the 2009 Pontiac G6 scanned by AmFam. Pre-crash steering was decoded but was not shown on the report. Checked the source and corrected the error that made pre-crash steering disappear.

Noticed that the Pontiac had steering wheel angle backwards as the convention. Instead of adding more code in the report HTML file to change the direction, I added a negative sign to flip the direction.

Checked the 2001 Lincoln LS scanned by HUB. The delta-V value was off by a little bit. Checked the source code and confirmed the processing code was correct. The difference comes from the conversion factor used by Bosch. The module records longitudinal acceleration in g, and the CDR converts it to delta-V in mph. However, instead of using the more accurate conversion factor of 1.60934, it just used 1.6, so the results are slightly different.

Went back to Android studio and switched the API route back to the original, as the v1 path might not be ready for a while.

Updated 'MyMultiObdCommand' to use vehicleGroup instead of make in determining whether to resend or not. Moved vehicleGroup enum into its own file so that it could be shared. This would take care of the Lamborghini issue as it is part of the Volkswagen group in the app.

Tested the app with Busmaster configured to cause the app to resend a command, using the Lamborghini VIN. The app worked as intended.

Made a list of endpoints and started thinking about how to actually secure them. For some pages that are already borrowing the template from the dashboard, it might be a good idea to just use the dashboard login page. With that, it might be possible to use a JWT, which would solve a lot of problems.

March 27 [8:30am - 1:30pm; 2:00pm - 5:00pm] [8 Hours]

Experimented with detecting incoming IPs in Elastic Beanstalk so that we can just lock down admin APIs using IPs. Tried a few things but the server would only see the proxy IPs. It would display the load balancer IP, plus 2 other IP addresses, but none of them were my actual IP addresses.

Looked into redirecting back to whatever page the user was on, if redirected to login. This means that, if a user clicks on a preview report link, the dashboard will redirect back to the preview report page after user logs in.

Replayed the 2013 Volvo S80 scanned by AmFam but CDR crashed again like it did for another Volvo in January. Tried the CDR900 and got the same error.

Went over the ISQ for AllState with Shekar.

Compared the trace from another Volvo with the 2013 S80 from today. Moved some responses to a different location within the replay file and the CDR worked. This might be a Busmaster file reading issue.

Checked the data and released the report for the 2013 Volvo S80.

March 28 [1:30pm - 3:30pm] [2 Hours]

Generated new numbers for 3rd party vehicle weights in claims report.

Class	Curb Weight	Fuel Tank Capacity	Fuel Weight
Compact Car	1300 kg	47 L	34.7 kg
Midsize Car	1600 kg	60 L	44.2 kg
Van/SUV/Light Truck	2200 kg	136 L	100.4 kg
Full Size Truck/SUV	2600 kg	98 L	72.2 kg

For each class of vehicles, a driver weighing 80 kg was assumed. The final numbers were rounded up from these estimates.

Added example vehicle names in the report for these assumed class of vehicles.

March 30 [8:30am - 1:30pm; 2:00pm - 5:00pm] [8 Hours]

Continued with security enhancement. Modified 'login.js' to redirect back to whatever page it was on. Tried 'referrer' with little success and apparently it would not work if user accessed the site from a bookmark instead of the URL bar. Ended up using local storage to store the origin URL. If login succeeds, 'login.js' will redirect back to the originating page.

'preview_report.html' was using Jinja2 to fill in some text. This meant that if user blocked a redirect by pressing 'Esc', the content would be in plain sight, defeating the purpose of requiring logins.

Modified 'preview_report.html' to only show the title and section headings by default. The page then checks the login status, and will redirect to the login page if necessary. Upon returning, the page will get a separate API to get the report preview text and set the page title.

Checked and released a 2007 Lexus IS from last Friday. Updated occupancy classification for Toyota.

Experimented with having the 'login' endpoint set an HTTPOnly cookie containing the access token. Subsequent API calls thus would have the token in the cookie for the server to verify.

March 31 [8:30am - 1:30pm; 2:00pm - 7:00pm] [10 Hours]

Looked into a 2012 Chevrolet Colorado scan that had only pre-crash data with no delta-V. This report falls between having crash data and having no crash data, so some of the code did not work as intended.

Modified the report text logic to be less strict, so that this scan would qualify as having crash data but with LVI. Moved the EDR trigger threshold text to a separate function for better organization.

Looked into 504 Gateway Timeout on AWS. Searched on Google and found that the timeout could be increased in Elastic Load Balancer. Changed the timeout for both the dev and prod environment to 600 seconds, from the original 60 seconds. This should significantly help with MG report generation.

Checked and replayed a 2018 Volkswagen Tiguan. This vehicle had all 6 event slots filled, but 4 of which had only pre-crash data, with all 0 delta-V values. Modified the logic to display an event as 'Pre-Crash Data Only' if all delta-V values are 0.

Noticed in the server logs that there were HTTP 408 errors, possibly due to the change in idle timeout of the load balancers. Changed the timeout back to 60 on the dev environment and the error went away. Since Elastic Beanstalk health monitoring ignores 4xx errors, this is okay. It is probably better to let the production server be capable of running MG reports with the 408 errors as a side effect.

Emailed Michael back about server IP and vulnerability patches.

Checked and released a 2005 Toyota Tacoma report after updating the resolution in the DB.

Researched into getting client IP address again. Consulted with Manjeet and noticed that switching off proxy in CloudFlare would make my actual IP accessible in Python / Flask. However, he also mentioned that it would be less secure because the server's EC2 instance would also be exposed to the public, instead of hiding behind a firewall from CloudFlare.

In the HTTP headers, the field 'CF-connecting-IP' was supposed to contain the client IP, but CloudFlare kept showing a random IPv4 address which was not my actual IP. Went through CloudFlare dashboard and found a setting called 'Pseudo IPv4'. This setting was enabled by

someone three years ago, and it would automatically select an IPv4 address if the client only has IPv6 address, which was the case with my laptop. Once the setting was changed from 'Overwrite' to 'Add', 'CF-connecting-IP' would show the actual client IP address, as either IPv4 or IPv6.

INVOICE

INVOICE #	20200504		DATE	2020/05/04
MAILING INFO	**Brian Chang-Yun Hsu**		BILL TO	**Collision Sciences Inc.**
	324-36 Via Bagnato			192-1055 Shawnmarr Road
	Toronto, ON M6A 0B7			Mississauga, ON L5H 3V2
PHONE	(647)-898-4560		PHONE	(905)-599-9899
EMAIL	changyun.hsu@gmail.com			

Date	Description	Hours	Rate	Amount
2020/04/16		7.0	$41.00	$287.00
2020/04/17		8.0	$41.00	$328.00
2020/04/18		0.0	$41.00	$0.00
2020/04/19		2.0	$41.00	$82.00
2020/04/20		9.0	$41.00	$369.00
2020/04/21		9.0	$41.00	$369.00
2020/04/22		9.0	$41.00	$369.00
2020/04/23	See Attachment for Details	9.0	$41.00	$369.00
2020/04/24		8.0	$41.00	$328.00
2020/04/25		0.0	$41.00	$0.00
2020/04/26		2.0	$41.00	$82.00
2020/04/27		8.0	$41.00	$328.00
2020/04/28		9.0	$41.00	$369.00
2020/04/29		9.0	$41.00	$369.00
2020/04/30		7.0	$41.00	$287.00
		96.0	**SUBTOTAL**	**$3,936.00**
			TAX RATE	**13.00%**
			TAX	**$511.68**
			TOTAL	**$4,447.68**

Hrg. EX.
072

Brian Chang-Yun Hsu Work Log
April 16, 2020 to April 30, 2020

April 16 [9:00am - 1:30pm; 2:00pm - 4:30pm] [7 Hours]

Upgraded the instance type for the dev environment to t2.small. Deployed code that would install the Armor agent in both environments.

Replayed and released a 2007 Pontiac Grand Prix for AmFam.

One Street Delivery user accidentally selected 'Street Delivery' in the client selection dialog. Removed that entry in the database so that AmFam would be the only option.

Updated the CrashScan setup guide to include Subaru and updated the company website address. Adjusted the layout a bit.

Continued working on automated invoicing. Created a new endpoint, 'send_invoice', that would generate a management report and an invoice, and would append both files to an email.

April 17 [8:30am - 1:30pm; 2:00pm - 4:00pm; 8:00pm - 9:00pm] [8 Hours]

Updated the link expiration page to include an input box using Bootstrap CSS. Created a page for when link renewal request is authorized.

Added new route 'info' to host static pages. The 'api' route should only host actual APIs.

Registered users from Copart.

Got a 2012 Acura TL scan from AmFam. This module does not support steering wheel angles (CDR report did not show them), but our code does not know to skip processing.

Found the bytes that indicate the EDR version for Honda. Now version 0.0.1.0 is coded to skip processing for pre-crash steering. All other versions will attempt to process pre-crash steering. Released the Acura report.

Replayed a 2017 Kia Rio. As seen before, some of the Kia reports generated using the Hyundai tool would look wrong. Our report looked more reasonable. Released the report without modification.

Finished the frontend pages for link renewal. There is a main page that allows user to enter an email address, and then there are two pages for success and failure messages respectively.

Got started making code for determining the details of the expired link. The goal is for the backend to know what exactly was expired just by decoding the JWT, so that it would know what to resend.

Checked and released a 2002 Chevrolet Tahoe report.

Added more code in the new 'renew_report' endpoint for each type of links (claims, management, and invoice)

April 19 [1:00pm - 3:00pm] [2 Hours]

Made a new email template for link renewal. The template will take the color code for the button from the Python code so that report has its regular blue button and invoice has its regular green button.

Finished up the new 'renew_link' API. This is the API that gets called from the 'link_expired' page. It receives the link ID (token) and checks its validity. If everything checks out, it calls the 'send_link_renewal_email' function.

The 'send_link_renewal_email' function writes a new expiration time to the existing JSON payload and encode it back to JWT. Then, it creates a link for 'create_report' or 'create_invoice' depending on the document type. For now, it is assumed that:

1. Preview report link does not expire. Users need to log in using the dashboard login page.
2. No management report link will be released to clients. The URLs for a particular month in PDF management report will be removed once the dashboard allows access to the same information.

This essentially means that the only report link that can be renewed if a 'create_report' link.

April 20 [8:00am - 1:00pm; 1:30pm - 5:30pm] [9 Hours]

Looked into the feature request from Copart. Essentially, to determine if a particular feature is present, CAN IDs may be used. However, a feature might be embedded inside a typical module and thus knowing the CAN ID might not be sufficient. This also needs a ton of reverse engineering.

Added 'reporting period', 'number of vehicles scanned', and 'total fee' to the vehicle scan page.

Removed an error checking added by Xian for creating data tables. The code was basically preventing a table from being created if it has no data rows. However, the table should be rendered with a default message of 'No data available'. The error checking code caused the JavaScript code to fail.

Modified 'get_scans_by_email' API to optionally receive a parameter, month, in the format of MM-YY. This can then be used to query the database for data or scans within that month.

Modified the current 'get_start_date' and 'get_end_date' functions so that they would be able to receive the 'MM-YY' string and returns appropriate dates.

Checked and released a 2016 Toyota Tacoma report.

Added the claim number for an AmFam case where the user accidentally selected 'Personal Vehicle'. Released the PDF report.

Added a new API 'get_date_options' that returns a list of dates that would be shown in the dropdown menu on the vehicle scans page. For now, the list is hardcoded in Python to be from December 2019 to this month.

Modified 'vehicle_scans.js' to call the new 'get_date_options' API and populate the dropdown list.

Got a 2000 Jeep Grand Cherokee scan from Street Delivery. The vehicle is unsupported but the report failed to recognized that. Added additional checks in 'report.py' for scans without a CrashScan row.

Finished up the dashboard page for vehicle scans.

April 21 [9:00am - 1:00pm; 1:30pm - 5:30pm; 9:00pm - 10:00pm] [9 Hours]

Fixed the position of copyright information on the dashboard.

Updated the content of invoice email to include additional information. Updated management report Python code to provide the additional information.

Checked and released a 2001 Chevrolet Silverado. This is another pre-crash data only scan.

Fixed the date issues with duplicate scans. Previously, if a vehicle has duplicate scans and the scans fall under different months, management report would display the latest scan date even if it falls outside the specified date range.

Implemented hyperlinks for sending invoices on the 'global revenue' page. These links are unprotected right now.

Experimented with Xero's 'offline access'. Created a new table to store access token and refresh token from Xero. This should allow the backend to always have access to Xero without user intervention.

Checked and reviewed a 2017 RAM 2500 report. The passenger seat belt was not actually configured. Added checks for driver and passenger seat belt configuration in Python. If the switch status is not configured, the status would show up as 'Unavailable' in the claims report.

Continued working on 'send_invoice' workflow by adding API call to Xero directly. Now the 'send_invoice' API does the following:

1. Generates management for the previous month for the client. The month cannot be selected as of now.
2. Generates the invoice for the generated management report
3. Email both files as attachment to the primary contact of the client
4. Call Xero API and POST a new invoice for the previous month

Modified the 'global revenue' page to send invoices by calling a JavaScript function, which can call the 'send_invoice' endpoint with an access token. The hyperlinks are not actual hyperlinks. The page uses the <a> tag only for a consistent look.

April 22 [8:00am - 1:00pm; 1:30pm - 5:30pm] [9 Hours]

Discussed with Kevin details on iOS app. Made a few suggestions for improvement for the UI of ClientSelection.

Realized that claim info dialog actually does not take into account the speed limit unit right now. It is okay since no US customer requires speed limit entry at this point. However, with the API-controlled claim info dialog being available, we might accidentally allow that while forgetting about the units.

Added 'speed_unit' in the claim info database table so that apps will be able to send that information. The 'speed_unit' value comes from the 'get_app_config' endpoint, called when the app first launches. The value returned by 'get_app_config' is determined by the 'get_units_by_country' function in 'processor.py'. This means that we only need to specify the country of the organization and the rest would be automatically determined.

Updated the 'get_claim_info' endpoint because the format of the response was not set up properly. This endpoint is used by the 'enter_claim_info' page, which has been updated to work with the new format. The other frontend app that uses this endpoint is the Android app. This is a convenience feature and could be disabled until the 'v1' endpoints are up and running.

Tested with the current Android app and confirmed that the new endpoint will not cause the app to crash. The app would just skip and not pull any existing claim information into the app.

Updated relevant Android code for claim info to work with the new response format for 'get_claim_info'.

Modified the Android app to receive the speed unit from appConfig and display in the claim info dialog if the speed limit is configured for that organization.

Noticed that email was not sending out property due to the recent change in email template to take banner images from the static folder. Updated the code and restored emails. Resent 5 scan emails manually.

Revised the post-processing code for CrashScan data. The code will now prioritize 'no response' over other kinds of blocking criteria like model year > 2020.

Updated the callback function code for 'global_revenue' so that the hyperlinks do not need to have IDs assigned and no loops would be needed to assign the callback. Now they all belong to a class 'send_invoice' with value being the organization ID and title being the organization name.

Added popup using Magnific Popup that Xian already imported. This provides more visual feedback when the API call to 'send_invoice' is processing.

April 23 [9:00am - 1:00pm; 2:00pm - 6:00pm; 8:00pm - 9:00pm] [9 Hours]

Updated 'create_report' endpoints in 'v1' to accept token in headers as well. This allows one-click access to reports for users (the token will show up as token={token}) in the URL. At the same time,

the endpoint will also allow API calls from either apps or dashboard that already has an access token.

If access token in header as ['x-access-token']:

1. We know the user because user_id is encoded in the token
2. We do not know what report the user wants, so the normal query strings ('id' for base 64 encoded VIN, 'type' for report type, and 'country' for the country format)

If no token in header but 'token' is present as a query string:

1. Decode the token as JWT
2. If the token is invalid (made up by an attacker), redirect to that 401 page, using 'abort'.
3. If the token is expired, redirect to the 'link_expired' page.
4. If everything looks good, generate the report as usual.

Did a small test to see the behavior of the JWT package when decoding an invalid token (made up by an attacker, for example) v.s. an expired token. The made-up token raised an 'InvalidSignatureError' exception while the expired token raised an 'ExpiredSignatureError' exception. This distinction is important as the server needs to know when to redirect to the 'link_expired' page.

Added the @token_required decorator on the 'v1' view_report as a start to test having these endpoints accept either an access token with query strings or a report token.

Got a 2013 Kia Soul scan. The Hyundai tool could not make a report for it. Released the report since it looked reasonable.

Got a 2013 Ford Taurus report. The pre-crash data circular buffer offset appeared to be off, as the data values were correct but in the wrong order. Added its software number as an exception throughout the code as needed.

Corrected some claim numbers for HUB. Emailed George about claim number formats.

Continued working on the 'global_revenue' page for getting hyperlinks to call our API with access token in the header.

Reviewed and released a 2014 Toyota Tacoma report purchased by Edward Ingram.

April 24 [9:00am - 1:00pm; 2:00pm - 6:00pm] [8 Hours]

Determined that one-click document access using 'x-access-token' would just be too much work and complication. Changed the strategy and make these hyperlinks with tokens instead. These tokens, however, expire after just 40 minutes, instead of 10 days like the regular reports links for clients.

Created a centralized token generator in 'processor.py'. All functions would get a token by indicating the origin (API, App, Dashboard, Link Renewal Email...etc.) and the generator would provide an appropriate token.

Checked and released a 2013 Ford Focus report for AmFam.

Finalized the implementation of 'token_required' for report links on the dashboard. This is implemented for the 'view_report' and 'create_report' endpoints in v1.py. Admin endpoints are unsecured for now.

Tested IP-restriction for admin endpoint access again, using the pending report endpoint as an example.

Updated the 'get_report_fee' function to return the fee as well as the currency. Updated all calls to this function to handle the additional returned value.

Organized the conversion between country code (CA), currency code (CAD), and currency display text (CAD $). Added functions in processor.py as needed.

Manually added some missing fees and currency codes in the ReportPurchase table.

Checked and released a 2015 Buick Enclave report for AmFam.

Found out that the callback for Stripe was not writing the report fee and amount to the database. Checked the data returned by Stripe again and found the error. Updated the code.

Realized that Stripe would be unable to call our API after a transaction is complete once the token requirement is implemented. Separated out the code for Stripe to a different endpoint. Now we just need to update the endpoint on Stripe dashboard.

Restored the order of scans for AmFam back to chronological, but the delta-V values are still displayed for reference.

Updated the design of the internal server error page.

Found an error in the link renewal workflow. Corrected the issue and deployed to dev.

April 26 [12:00pm - 1:00pm; 3:00pm - 4:00pm] [2 Hours]

Tested the US East environment set up by Shekar. The environment itself was okay and the API used for Elastic Beanstalk health monitoring was responding, but anything that had to do with the database was not working. The environment probably needed another security group for connecting to the database.

Talked to Shekar and showed him on TeamViewer the deployment code that Eric and I used to setup the current Canadian environments. Added Shekar to Bitbucket for Python code access.

Searched for all occurrences of 'create_invoice' and wrote the new version with token underneath the current code. When ready, the old and new code can just be switched and all outgoing links would have the new format, linking to the new endpoints with token protection.

April 27 [9:00am - 1:00pm; 1:30pm - 5:30pm] [8 Hours]

If a user logs out from any page and logs right back in, the dashboard will redirect to the 'Overview' page. Removed all occurrences of 'redirectToOverview' as that page should not be accessible.

Commented out code in 15 dashboard HTML files to disable the circular icons on the top of the page. The only icon that works is the logout one.

Added more plans to the dashboard brainstorming page for vehicle scans.

Added invoice status for Xero to make invoices 'Authorized'. Replaced the contact ID with actual Xero contact ID from our database.

Noticed that authorized invoices cannot be deleted from Xero and can only be voided. However, even when an invoice is voided, its name is still used and thus a new one cannot be created. It is therefore best to leave the invoices as drafts as they can be deleted if needed.

Converted all 'admin' endpoints to use either IP restriction or link token. The endpoints are backward compatible so that links in old emails would still work.

For these 'admin' endpoints, the primary defense is the IP address. If the call came from a whitelisted IP address, the report would generate even if the link token had actually expired. This allows Theuns to view the report using the 'admin/view_report' endpoint right after a scan, with an expiration time of 10 days. After 10 days, if one of us wants to view a report at home, the IP address will allow us to bypass the expiration so that we do not have to constantly get new links through the 'link_expired' workflow.

Updated URLs in regional management report files.

April 28 [9:00am - 1:00pm; 1:30pm - 6:30pm] [9 Hours]

Looked into issues with access for Theuns. He tried to access the 'enter additional info' page but got a 401 error. Checked his IP and it actually changed. Googled his ISP and apparently they do not provide static IP for residential clients, so we will need some sort of solution for this.

Tried to generate a replay report for a 2020 Ford Explorer as HUB asked for it. CDR wanted 6 more new PIDs. Made up the response by copying from other part number responses and it worked. Replaced all responses with 0x00 and generated the final report.

Added the 4 new Hyundai cables in the new vehicle support Google spreadsheet.

Checked a 2007 Dodge Caliber report as it had a very high delta-V value. This vehicle only stores acceleration in g, and the server computes the delta-V using the g values. Copied some of the values into Excel and verified that the delta-V would be accurate. Released the report.

Added phone numbers of Street Delivery and AmFam users.

Added a new endpoint to support the dashboard in terms of menu options. This will allow the server to control the links seen by a particular user. For example, we definitely do not want client admin or user to see our 'global revenue' page.

The first step is to configure the dashboard to receive the menu options. The server can check the user and generate the option JSON based on the user's organization and role. The second step would be to add some sort of checks so that if the user hacks the system by entering a restricted URL directly, the dashboard or the server would be able to display a 401 Unauthorized page.

Talked to Shekar about JWT content because the token we use is actually just Base 64 encoded. While JWT has signature that will allow the server to validate it, it does not prevent a curious user from decoding the token and see his/her user ID used by our internal system.

Added a new column, 'session token', that is also a UUID, in the user database table. This is essentially an external, temporary user ID that can be used to point to a specific user. The server can invalidate it and replace with something else without affecting our internal records. Updated token generation code as well as token checking code to use the 'session token' instead of 'user ID'.

Updated endpoints used by the dashboard to stop using user emails in API calls. The access token now contains the session token, which will then allow the server to know exactly who the user is.

Noticed that the menu in dashboard would not expand on nearly all pages and it appeared a bit random. The randomness in this case suggested that it might be timing related. Analyzed the difference between various aspects of the old code and the new code.

It turns out that the 'theme.js' file applies the onclick listener code to those navigation items. Therefore, it does not know whether the 'menu.js' function finished getting and building the menu items. In this case, 'theme.js' tried to look for navigation controls but the API had not yet been called. Once the API returns and 'menu.js' builds the navigation controls, 'theme.js' had already run through and thus the onclick listener code never got added to the controls.

Moved the code that applied onclick listener to inside 'menu.js', where it will run after the API returns.

April 29 [8:30am - 1:30pm; 2:00pm - 5:00pm; 7:00pm - 8:00pm] [9 Hours]

Modified the 'overview' page to include actual user information.

Found a simple user icon in SVG online and updated the placeholder user icon.

Updated the time on that page to be the server time (UTC). Added number of scans for the user. This number changes depending on the user role. For super admin, the count is for all scans. For admin, the count is for all scans in that organization. For users, the count is for all scans done by the user.

Checked Firebase and confirmed that our migration was okay and complete.

Updated the claim number for a 2018 Chevrolet Malibu after the investigator selected 'Personal Vehicle' again. Disabled that checkbox from the database.

Got a call from George White about the 2020 Ford Explorer. Generated another Bosch report with the same date as the claims report.

Updated the GCNA billing rule to include "no accident data" reports if they buy it anyways. Modified the report fee to be based on the standard 300 USD. The previously written 'convert_report_fee' will take care of the conversion. Python then rounds the result to the nearest $5.

Checked and released a 2020 Chevrolet Silverado report.

Filed a support ticket with AWS support to see if there is a way to manage multiple databases at the same time.

Got a 2003 Buick Rendezvous from AmFam. This is a cable 829 module that I had not decoded due to lack of data (we only had one report before this). CDR replay worked and it was a pre-crash data only scan.

Decoded available information and added in Python. For now, delta-V cannot be determined. Neither of the two reports we have had any delta-V tables. Released the report after deploying to production.

Checked and released a 2007 Toyota Tundra report, scanned by Oscar from Street Delivery after failing multiple times.

Went back and located the byte that controls whether pre-crash brake values were valid or not for GM VPW cable 829 subtype 4. Updated the Python code to check that byte.

April 30 [8:30am - 1:30pm; 2:00pm - 4:00pm] [7 Hours]

Worked with Shekar and tested the new US development environment. The connection to the database was working, but a table needed updating.

Added the static folder path in the AWS console for the US dev environment. Tested the 'preview report' page and kept getting 'connection refused' error. It turned out that this new environment did not have a classic load balancer, and I think it was not redirecting to HTTPS automatically.

Provided feedback to Shekar and we might end up using Eric's old code to make a US production environment. That way the database name can be modified.

Updated an image on the 'overview' page of the dashboard.

Added a new 'List of Purchased Reports' in the regional management report. For now, the 'List of Pending Payments' and 'List of Purchased Reports' would have some overlapping.

Created a new experimental 'clients' page that is a hybrid between 'vehicle scan' and 'global revenue'. Essentially, this is a page that lists the revenue from each client, and the page has a

dropdown to select the past 6 months. Additional logic could be added to hide the 'Send Invoice' button / hyperlink for the current month, because invoice should be sent only for past complete months.

Wrote a small script that extracted OBDLink serial numbers from scan JSONs. Copied and pasted serial numbers into the user Google spreadsheet.

INVOICE

INVOICE #	20200518	DATE	2020/05/18
MAILING INFO	**Brian Chang-Yun Hsu**	BILL TO	**Collision Sciences Inc.**
	324-36 Via Bagnato		192-1055 Shawnmarr Road
	Toronto, ON M6A 0B7		Mississauga, ON L5H 3V2
PHONE	(647)-898-4560	PHONE	(905)-599-9899
EMAIL	changyun.hsu@gmail.com		

Date	Description	Hours	Rate	Amount
2020/05/01		8.0	$41.00	$328.00
2020/05/02		2.0	$41.00	$82.00
2020/05/03		3.0	$41.00	$123.00
2020/05/04		8.0	$41.00	$328.00
2020/05/05		9.0	$41.00	$369.00
2020/05/06		9.0	$41.00	$369.00
2020/05/07		9.0	$41.00	$369.00
2020/05/08	See Attachment for Details	9.0	$41.00	$369.00
2020/05/09		0.0	$41.00	$0.00
2020/05/10		2.0	$41.00	$82.00
2020/05/11		10.0	$41.00	$410.00
2020/05/12		8.0	$41.00	$328.00
2020/05/13		8.0	$41.00	$328.00
2020/05/14		8.0	$41.00	$328.00
2020/05/15		7.0	$41.00	$287.00
		100.0	**SUBTOTAL**	**$4,100.00**
			TAX RATE	**13.00%**
			TAX	**$533.00**
			TOTAL	**$4,633.00**

Hrg. EX.
073

CS00510310

Brian Chang-Yun Hsu Work Log

May 1, 2020 to May 15, 2020

May 1 [9:00am - 1:30pm; 2:00pm - 5:30pm] [8 Hours]

Got a 2014 Honda Odyssey scan from AmFam, but the pre-crash steering was not shown in the CDR report. The raw hex bytes were all 0s so the claims report originally had the angles shown as all 0s. The same happened on a 2012 Acura TL two weeks ago, but this Odyssey uses CAN instead of K-Line.

Found a byte close to the pre-crash data section that directly controls the pre-crash data. It is probably used to indicate whether a specific data set (vehicle speed, engine RPM, steering...etc.) is supported or valid. If not supported or not valid, the CDR just does not show that column.

Found another byte responsible for indicating whether ABS and ESC are supported.

Binary	Effect
0000 0100	Brake
0000 1000	Accelerator Pedal
0001 0000	Engine RPM
0010 0000	Steering Wheel Angle
0100 0000	PCM Derived Accelerator Pedal
1000 0000	Vehicle Speed

Binary	Effect
0000 0001	Brake On/Off
0000 0100	ABS
0001 0000	ESC
0010 0000	ESC

There should something similar in Honda K-Line modules as well and should be revisited later. K-Line data has checksum so manipulating data takes a bit more work.

Discussed API penetration test with Michael and Taylor. Updated the documentation sent to Richard at Packetlabs and sent to Taylor.

Updated the experimental 'clients' page in the dashboard to link to the vehicle scans page. Modified the vehicle scans page to take additional query strings.

Updated the logic in 'get_vehicle_scans' to handle the three types of scans: scans done by self, scans done for others, scans done for self.

Corrected an issue in the 'v1/view_report' endpoint that failed to grab the start and end date for management reports.

Updated logic for HUB in 'management_report.py' to only show scans belonging to HUB. Scans done by HUB for AmFam are not currently shown since they are not relevant for billing purposes.

Got on a call with Kevin and discussed upcoming tasks for iOS.

Checked and released a 2008 Toyota Prius report and a 2016 Nissan Rogue report.

May 2 [12:00pm - 2:00pm] [2 Hours]

Updated the width of the Select2 dropdown box in 'vehicle_scan' and 'global_revenue' page. Tried a couple options, including auto width, but none worked as intended. Ended up fixing the width to 30%.

Updated the logic in 'vehicle_scans.js' to optionally receive the organization key so that a super admin can get scans for a particular organization.

Replayed and released a 2008 Scion tC report.

May 3 [11:00am - 2:00pm] [3 Hours]

Went back to Honda K-Line decoding and located bytes that were relevant for controlling pre-crash data display by systematically making data bytes 0x00. Found the bits that control pre-crash steering, ABS, and ESC.

Added checks for support for steering, ABS, and ESC in the Honda K-Line processing code in Python.

May 4 [9:00am - 1:00pm; 1:30pm - 5:30pm] [8 Hours]

Added code to disable the date selection dropdown in the 'clients' page when the page is fetching data from the server.

Noticed that total fee values were off slightly between the 'clients' page and the 'vehicle_scans' page. Updated Python code to not round fees when performing conversion. Rounding down to 2 decimal places should be done right before displaying the value on a page.

Updated the 'preview_report' page to use magnific popup instead of the built-in browser dialog.

Removed inline CSS styling code from 'preview_report' and 'enter_claim_info' pages. Inline CSS is considered insecure and it is better to move code into a CSS file. Added styling code in 'theme.css'.

Noticed that a HUB user was on iOS but using the 'hubent' account. That would explain why the AmFam claim number was not entered at runtime. Emailed George White and explained the situation.

Added hyperlink underneath claim numbers to allow a user or admin to update the claim number. Figured out a way to access the data in the table using built-in functions and APIs of DataTables. Removed the custom 'value' and 'info' fields from the hyperlink element.

Figured out how to update the claim number value in the table directly without reloading the page. The value would be updated if the server returns a success flag for 'send_claim_info'. This means the server successfully updated the number in the database, so that the frontend webpage is allowed to update its number as well.

Added an info popup to notify users whether the claim number update was successful.

Added a confirmation popup when 'send invoice' is clicked. Added code to grey out 'send invoice' hyperlinks for the current month.

May 5 [8:30am - 1:30pm; 2:00pm - 5:00pm; 7:00pm - 8:00pm] [9 Hours]

Pulled the latest iOS code from Kevin and tested a few combinations of organization and client selection. Noted quite a few bugs and wrote back to Kevin.

- The code did not build because representedClients assignment was wrong
- Client selection dialog title and search hint needed fixing
- Claim number hint was not configured based on received appConfig
- Claim number format was not configured, meaning error checking was done using the wrong criteria
- Old issue of the app not actually logging in. This means the app updates available popup is only available upon a fresh install, which completely defeats the purpose of the dialog

Got a 2005 Chevrolet Impala from AmFam. VPW replay did not work as CDR rejected the response to PID 0x2D. Checked the raw data and the module gave a positive response with all 0s. Typically, for this PID, the number of bytes in the response would be 2 bytes, but this module gave out 6 bytes of 0x00s. Modified the raw JSON data so that each response contains the correct number of bytes, and CDR generated a 'no data' report.

Modified the GM processing code to check for pre-crash data if delta-V is 0. Due to previous issues, the code started to allow delta-V of 0 because those scans had pre-crash data. This scan has no pre-crash data, so an additional check is put in place to mark a scan as 'no data' if delta-V is 0 and there is no pre-crash data.

Released the Impala report as well as a 2013 Nissan Maxima report after making changes to allow Nissan to have 0 delta-V as well. This is okay because Nissan uses 0x7F for null.

Called Kevin on WhatsApp and went over the issues I found today. Also talked about upcoming changes like the module detection dialog, as well as the transition to move away from using organization ID. He mentioned that it was really difficult to make the claim info popup scrollable. I

told him that if it is easier, the popup really should be made into a full-screen form like Android. That should be a lot easier in UI design and hopefully he would be able to figure out the scrollview stuff.

Tested the iOS code for access token. After correcting two small bugs, the API calls to v1 endpoints were working perfectly. The refresh mechanism can be left until later. In the worst case, we can just have the access token expire after one week so that users who scan will never run into token expiration issues.c

Looked into questions on a 2013 Nissan Juke's pre-crash brake status from HUB.

Modified the backend code to remove the use of 'organization_id' in v1 API endpoints. Added if statements through the backend code to use 'organization_key' if present in the incoming data.

Did a test scan using the current Android app on the dev environment and confirmed that the new code would not interfere with existing workflow.

May 6 [7:00am - 1:00pm; 1:30pm - 4:30pm] [9 Hours]

Removed usage of 'retrieve_report' endpoints from the Android app because that endpoint is deleted in V1.

Checked the iOS app for usage of 'retrieve_report'. The source code did not contain any occurrence of that keyword. The app also appeared to have the button hidden for all users.

Updated the data retrieval logic in 'get_vehicle_scans' to first get all scans done by users in the company and then get all scans done for the company. This logic is borrowed from the 'get_client_info' endpoint.

Modified 'management_report.py' so that HUB's report would show all scans done by their users, but with fees for HE reports.

Got a 2016 Audi A5 scan from Recon Engineering. CDR showed one event but all data values were invalid. The engineer had the CDR and his report said the same thing.

Checked the raw data bytes and saw they were all 0xFEs. Made 0xFE in addition to 0xFF to be invalid in the processing code. The code will produce a 'no crash data' report.

Continued working on refining the 'clients' page. Added a new column to show the number of purchased reports in addition to the vehicle scans column.

Modified the 'purchased reports' page to accept links in similar format as 'vehicle scans'.

Added a new sample API 'get_vehicle_data' for marketing purposes. The endpoint returns the text generated on the report preview page, but in plain text without HTML formatting. The endpoint is restricted to only 3 VINs. The list can be modified as needed.

Modified the previously written API documentation for this new sample endpoint.

May 7 [8:00am - 1:00pm; 1:30pm - 5:30pm] [9 Hours]

Registered Robert from 360 Accident Investigations.

Checked the server log for the failed scan by OM. The scan got CAN error, which sounded like a power issue as Drive Clean also failed. Notified Theuns.

The 2018 Ford Ranger from OM cannot be replayed, but the data interpreted by the Python code appeared okay.

The Hyundai i20 was replayed back to the Hyundai tool with 'Kona' selected as the model. The Hyundai tool again presented some mismatched data in the pre-crash data section, which also happens when trying to replay a Kia scan to the tool sometimes. The interpretation from my Python code made more sense and the report can be released.

The failed Volkswagen Polo was rescanned 4 hours after the initial scan and it worked perfectly this time. Released the report.

Found a bug that caused popup callbacks to be run twice on the 'clients' page. This would cause two management report / invoice emails to be sent to the same client. The cause was that only a portion of the page gets rebuilt when selecting a different month, and the portion that stays gets a new callback attached every time the page rebuilds.

Moved the code to outside the table building code but that did not work either. It seems that onclick listeners need to be reassigned when the table is recreated, and must be removed when the table is removed. Added code in the current 'resetUI' function to remove onclick listeners.

Went back to 'vehicle scans' page as it too had issues with regenerating the table. The onclick listener seemed to stay and the built-in destroy() function did not remove it. Found a post on Google that adding '.DT' to the click event, making it 'click.DT', would cause DataTables to automatically remove the listener. This is currently the only solution that works.

Checked and released reports for 2006 Toyota Highlander (on 15-minute K-Line) and 2004 Cadillac Escalade (on VPW).

Added several debug statements and figured out how and why onclick events in 'tr' and 'td' elements were triggered. Added stopPropagation() in the 'td' onclick callback so that it does not trigger another 'tr' event.

Added a light gray circuit diagram pattern to the background of the dashboard.

May 8 [8:00am - 1:00pm; 1:30pm - 5:30pm] [9 Hours]

Downloaded and installed CDR version 19.4. Used a 2017 Subaru Crosstrek VIN with data from the Stark 2017 Subaru Outback. The CDR ran the same process, which is kind of expected as the first thing it asked for was the module ID. Based on the module ID, it determines which process to run.

Checked a 2014 Jeep Patriot scanned by OM. The scan actually had some runtime errors and so five responses were missing data. Checked the server log and noticed that the app did not automatically rescan because the server returned 'Manual Decoding Required' for OM scans. Added an exception for when data is missing so that users get 'No Response from Vehicle' instead, which would make the Android app automatically rescan. Notified Theuns that this Jeep should be rescanned.

Checked a 2011 Kia Cerato scanned by Theuns. Technically, 2011 models are not supported, and this module is a variant that we had not seen before. Corrected the pre-crash engine RPM issue as the index was shifted by one for some data sets in this module. For other information like seat belt and airbag deployment, since the Hyundai tool would not accurately interpret the data, the report would not contain them.

Added new vehicles, supported regions, and cable numbers for the new entries in CDR version 19.4

Got the first scan done by Theuns so now we have a module ID to use with the CDR. When that module ID, the CDR started sending out different PIDs, although the security access algorithm appears to be the same.

Created a special version of the Subaru process that only gets triggered for Theuns' VIN. This version would ask for supported PIDs under 0x10, 0x20, 0x21, 0x22, and 0x24.

Got the second scan back from Theuns. However, during replay, CDR wanted a PID that the app did not ask, meaning it was likely that the PID was unsupported. Made a all PIDs version JSON file for this special Subaru process, where it would ask the car 22 {SID} XX, with XX going from 00 to FF. The total requests would be around 1280.

Checked a 2013 Jeep Wrangler scan. The driver seat belt was buckled but our code said 'Unavailable'. Checked a couple decoding reference documents and adjusted the logic to determine if seat belt switch is configured. In this case, the seat belt configuration did not need to be checked. Checking it would instead lead to 'Unavailable'. Released the report.

May 10 [9:00am - 11:00am] [2 Hours]

Ran a query for data on South Africa and Namibia scans. Generated a list of year, make, model, VIN, and pre-crash speeds for Theuns.

Theuns pointed out that a 2018 Ford Fiesta's pre-crash data was out of range. Checked the raw data and updated the cable number to 783. The change in cable caused another 2018 Ford Fiesta's pre-crash speed to get out of range. Compared the VINs of the two vehicles and realized that one of them was actually a 2013 model. Updated the VIN table as well as the CDRData table, then reverted the change of cable number back to 823. Regenerated the list.

Checked CDR version 19.4 for new Toyota part numbers. Wrote the 21 new part numbers into the database table.

May 11 [8:30am - 1:30pm; 2:00pm - 5:00pm; 7:00pm - 9:00pm] [10 Hours]

OM rescanned the 2014 Jeep Patriot. Checked the raw data and the scan had even more PIDs with missing data. However, between the two scans, there was enough data to piece together a JSON file for report generation.

Copied data from neighboring requests for the two missing responses. They were part of the pre-crash data set and FCA has 51 data points, so borrowing data from neighboring requests has minimal effect and the generated CDR report will not be released. The data matched but CDR did not report the lateral delta-V even though the module definitely had data for it.

Theuns pointed out a small error in the claims report because of the prior update of claim info speed limit unit. Added an if statement to check for null.

Got started working on a data conditioner for FCA based on the existing code from BMW. The goal of this conditioner is to resolve any data mismatch issues so that subsequent processing code can work under the assumption that no mismatch would occur.

Got a 2009 Jeep Wrangler scan from Street Delivery. The delta-V seems to be backwards in terms of the sign. The pre-crash brake was also not correct. Queried the database and found that there were no scans from cable 227 with crash data so far. Checked the raw hexadecimal data and adjusted the Python code to correct the sign. Also updated the index for pre-crash data. Deployed new code and released report.

Got another scan from Theuns. Fed the database back to the CDR and the process finally went through. The generated report has no crash data, which Theuns confirmed would be accurate.

Checked the PIDs asked by the CDR, the PIDs indicated as supported by the module, as well as the trace from the all PID scan done by Theuns. It turned out that the module would have positive response to PIDs that are not supposed to be supported. Furthermore, the CDR would sometimes ask for those "unsupported" PIDs. It is therefore too risky to assume the support information from the module can be trusted.

Created two new Subaru request JSON files. One for the older process with PID 0x10, 0x20, 0x21, 0x22, and 0x24, resulting in 1280 requests. The other one is for the newer process, with PID 0x10 and 0x23, resulting in 512 requests.

Noticed there were some vehicle_model_id mismatches between the VehicleModel table and the EDRSupport table because some vehicle models were added in advance from the South African vehicle list. Wrote a small script to unify the IDs between the two tables.

May 12 [8:30am - 1:30pm; 2:00pm - 5:00pm] [8 Hours]

Tested the Subaru process using different VINs to make sure the correct file gets read and used for the scan. Deleted the special case for Theuns' VIN. Archived the previously written Subaru process code.

Updated and filled in all missing cable numbers for Subaru. For South African region code entries, the cable number is assumed to be the same as North America. Made all Subaru models using cable 614 and 616 as supported in addition to the already supported cables 835 and 836.

Got a call from George White about an in-field issue for a 2001 Lexus 430 (I did not catch the exact model). After resetting the dongle and starting the engine, the scan (on iOS) still would not work. George told John the investigator to move on.

Looked into an issue with missing email. HUB did a scan yesterday on a 2017 Honda CR-V. No one got an email for that scan. Checked the AWS log and it looked like the app stopped half way. It called the API for enhanced diagnostics support, but did not continue to run it, and there was no API call for sending the email, either. Emailed George to see if he could ask the investigator if anything out of the ordinary happened during the scan. The person was using the latest Android app (v1.12).

Got a 2012 Dodge Grand Caravan scan that had some weird decoding issue. Did a CDR replay and the report came out clean. Checked the Python source code and corrected an error in the delta-V calculation code. With the correction, the code recognized the delta-V array was all 0xFFs and thus the module was actually empty. Released the report.

Registered another user under CSA. There are now 3 people who scan under CSA for other clients.

Got an email from Theuns regarding Telesure Investment Holdings and their claim number. Discussed with Theuns to settle on a simplified claim number format of 9 digits followed by 2 digits, #########/##. The 9-digit portion is the policy number and the 2-digit portion is the claim number. The claim type and the brand name is not included.

Added 'app_config' for TIH. Also added their organization ID as a represented client for CSA.

Added the domain for TIH to the database and set up the email rule so that TIH users do not get emails. Instead, CS admins would get an email. This is the same setup in use for all SA clients except OM iWYZE.

Registered in batch the users from TIH.

Replayed and checked a 2014 Toyota Prius scan. The EDR portion was okay, but there were two DTCs with unknown definition. Looked up on Google and added the definition to the lookup text file. This text file should be converted to a database table at some point.

Added preliminary code for a new 'Emissions DTC' table in 'models.py'. There should be an 'EnhancedDTC' table later to store enhanced diagnostics DTCs.

Checked and released report for a 2012 Dodge Charger.

May 13 [8:00am - 1:00pm; 1:30pm - 4:30pm] [8 Hours]

Discussed with Renan about a lab setup that would allow data retrieval from modules using the CrashScan app without the Bosch cables. The network probably would be powered by a 12V adapter like all setups we had so far. Then, from one of the female OBD end, have some sort of cable that connects the OBD network to the module. Made a simple diagram showing the setup.

Got a 2015 Ford Econoline scan from OM that did not have a cable assigned. Added an entry for this vehicle and assigned cable 384 to it. However, since it had 3 random non-0xFF bytes in the otherwise all 0xFF region, its response size was determined to be 2192. Modified the response size calculation code to allow a bit more flexibility and will only stop calculating when it sees two consecutive non-0xFF bytes. This made the scan come back with 1562 bytes, which we have seen before.

Talked to Theuns about this scan and he said he never heard of the model. Turned out that the vehicle was actually an EcoSport. Updated the entry in the 'vin' table as well as in the 'cdr_data' table. Removed the entry of the manually added Econoline for South Africa.

Wrote a small script to write DTC information to a brand new database table. Dealt with any duplicates in the file.

Modified 'drive_clean.py' to use the database table instead of the text file. That allows instant update of DTC information without deployment to AWS.

Read through the API pen test report and started to address some of the issues.

"mail.collisionsciences.ca" is used by the AWS environment for Mailgun. With Mailgun, it is possible to define a custom domain, so that emails would show up as signed by "mail.collisionsciences.ca", which would make our emails look more legit. The actual path leads to the old website, which has PHP scripts. PHP is out of date so we could either disconnect that page from the domain, or update PHP.

Talked to Manjeet and it did not seem like it would be easy to update PHP. He suggested creating another environment on AWS to host the latest version of PHP. However, since we do not even use the PHP site, creating an AWS environment seems unnecessary. It seems that taking down the old website might be the easier way to go.

Updated the cipher of the elastic load balancer in the EC2 instances. Based on suggestions from Michael and Skekar, only TLS version 1.2 is allowed.

Added input validation for VIN and country in Python. The VIN must be a string, must have 17 characters, and must be alphanumeric. This will prevent XSS as a script would contain symbols like < and >, which would fail the alphanumeric test. The country must be a string, must have 2 characters, and must be alphabetic.

Made a copy of the check_support page and deployed to AWS.

Updated a few API calls in iOS to include the country code as a query string.

May 14 [8:30am - 1:30pm; 2:00pm - 5:00pm] [8 Hours]

Read up on the Axe Android package as well as accessiBe. The Android package requires some programming on our end, and we do not know right now if all texts are supposed to have an audio backup. Accessibility requires images to have an alternative text or description. Images in our PDF reports are all decorative icons with not much meaning. It would be relatively simple to add the 'alt' attribute to those <img> tags. However, the text might not show up in the generated PDF and still needs to be tested later.

Registered another new user for Telesure Investment Holdings. This time Theuns got CC'd correctly.

Made a blank HTML page 'index.html', and added via FTP to the old website server. That essentially removed the old company website home page. The blank page has a permanent redirection URL to the new company website, which does not have PHP or WordPress.

Revisited the Flask-Talisman package to address some of the security header issues. By default, it adds Content Security Policy, which restricts loading external resources like Google Fonts and prevents inline CSS from working. Added code to override the header value to allow inline CSS, but that showed up in the browser, disclosing the fact that we allow inline CSS. Ended up assign an empty Python dictionary to that, which removed the header entirely.

Team CISO pointed out that a 'Cache-Control' header needed to be added, but the Talisman package did not have that option. Ended up removing the package entirely. Instead, using the 'after_request' decorator in Flask, a function can be written to run after each request / before sending back the response. The list of security headers are now added there, following the recommendation from Team CISO.

Manually added the claim number from the HUB scan of 2011 Lexus from yesterday.

Emailed back to Michael with a few questions on the vulnerabilities.

Replayed and released a 2015 Hyundai Sonata report for AmFam.

Added input validation for UUID using the built-in uuid Python package. This will check to make sure the input is a string and is a valid UUID. It does not check whether this UUID is a valid key for our application. An invalid UUID key would simply not match anything in our database.

Removed all occurrence of 'Access-Control-Allow-Origin' and CORS in the Python code. First tested to make sure the removal would not break anything then deleted all occurrences. Some of these were added before I had a local server running and some were added because Xian needed them.

Added input validation for Base64 encoded VINs. Added a try-except-else statement around the decoding function to prevent unhandled exceptions. If the decoding fails, the VIN returned would be NULL, and the API can then return a 400 error code.

Fixed the HTML for the 'vehicle support' page following advices from an automated accessibility scan.

May 15 [8:30am - 1:30pm; 2:00pm - 4:00pm] [7 Hours]

Added 'past_posting' back in the report as a condition to show the Low Velocity Impact section of the injury section. Previously, the logic was to show that section if no crash data was found or if the most recent delta-V is greater 8 km/h. In the case of the 2015 Hyundai Sonata, the most recent delta-V is less than 8 km/h, but it was also 4132 ignition cycles old, so the LVI section should be present.

Added security headers to all API routes.

Realized that 'Access-Control-Allow-Origin' would be needed for the vehicle support lookup API because the lookup page is hosted on Manjeet's server and that API call would cross origin. Tested to make sure that the API would indeed fail without 'Access-Control-Allow-Origin', and then added the header back to the API. Essentially, the two API endpoints for looking up vehicle support is left open to allow any origin.

Got a 2015 Subaru Outback scan with crash data. This is the newer cable that I had already decoded and not the batch recently added in version 19.4. Checked the report and corrected an error in the processing code. Released the report after deployment.

Checked a 2018 Ford Focus scanned by Theuns. The delta-V was 63 km/h and was confirmed by the CDR after using a North American VIN. ABS was engaged but pre-crash ABS had not been decoded yet. Added pre-crash ABS to the code.

Started analyzing the color contrast in the Android app. Since the app's background is a very dark blue, most texts would have enough contrast. The only part that failed to meet the AA standard was the check engine light on white background.

Experimented with some alternative design and added a black background to the two icons. Adjusted the spacing in the dialog to compensate for the change in the icons.

Got a call from George White about a 2015 BMW 5-Series that would not start despite having battery connected. The car's alarm was going off in the background so there should have been power. The investigator said the car did not have a key, and that a replacement was provided to him. Given the fact that the alarm was going off, it was likely that the key was invalid. Advised them to remove the airbag module if data is really needed.

Found a YouTube video for that generation of BMW 5-Series and emailed the link back to George.

Started analyzing the color contrast on the claims report. There were a couple of sections (white on light blue and green on white/light grey) that failed the test. Found alternative colors that would at least meet the AA standard.

Updated the light grey text and icon used to indicate irrelevant information in the claims report to be a darker grey.